DISCOVER THE WORLD OF RASPBERRY PI!

NEW MAG!

PiUser

YOUR NEW SOURCE FOR RASPBERRY PI

JOIN THE PI REVOLUTION

Issue 02 // Spring 2017

20 AMAZING RASPBERRY PI APPS

Discover the best web browser, media player, games and more

15

Pi projects to try

Q&A SPECIAL

ALL YOUR PI PROBLEMS SOLVED!

Revealed! Simple solutions to your most common Pi questions

Future

PIU02 2017

Make a Pi walkie-talkie

Retro-syle fun!

Build the ultimate Pi media server

The perfect starter project

Learn to code with our guides

Master Python and Scratch

SHOUT
SHARE
IT

72
techradar.

The source for tech buying advice

techradar.com

PiUser

YOUR NEW SOURCE FOR RASPBERRY PI
JOIN THE PI
REVOLUTION

Welcome!

We're here to inspire you to make and learn with your Raspberry Pi, so let's get started!

We had some great feedback to the first issue of Pi User. It's very encouraging to discover that there are so many enthusiastic Raspberry Pi users out there. And here's the good news: if you loved issue 1, then you'll love issue 2! We're back with some more great Pi projects involving home electronics, coding and ingenuity.

First, though, an apology. We made a mistake with colour choices last issue which rendered sections of code on a couple of tutorials illegible to many, so we've added the code in question to our Facebook page at **www.facebook.com/PiUser/**, saved it in a Zip file which you'll find at **http://bit.ly/Pi-User-01-code** and posted the text at **pastebin.com/MY9LGYFz**. Head to any of these and you can find all the sections of code you're looking for.

This issue we've got lots of treats in store, including the 20 best apps for your Pi. All this amazing software is ready and waiting for you to install right now – and best of all, it's free, so follow our guide and start getting more from your Pi today. We've also got a tutorials on making your own Pi walkie-talkie, Minecraft coding and creating the perfect media server on a Pi.

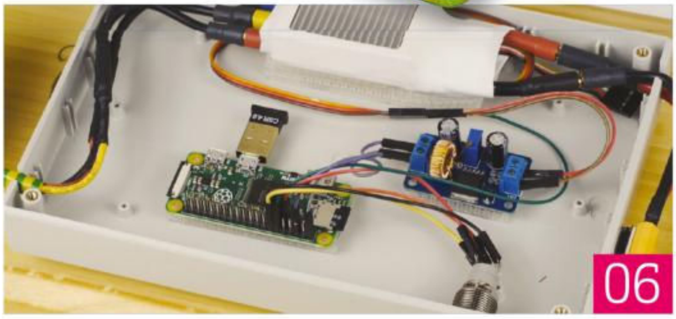
Enjoy the issue and let me know what you think. Mail me at **graham.barlow@futurenet.com** and find us on Facebook. I'll see you next issue, on sale 23 May.

Graham Barlow Editor-in-Chief

Contents

Build a SUPER Pi!
22

Beginners guide to RetroPie
35



WORLD OF Pi

Discover what a Pi can do!

06 Our favourite home Pi projects

11 The Top 20 Raspberry Pi apps

22 Power up! How to build a Super Pi

35 The Top 20 Raspberry Pi issues solved!

Pi PROJECTS

Have an adventure with your Raspberry Pi

48 Play Doom on your Raspberry Pi

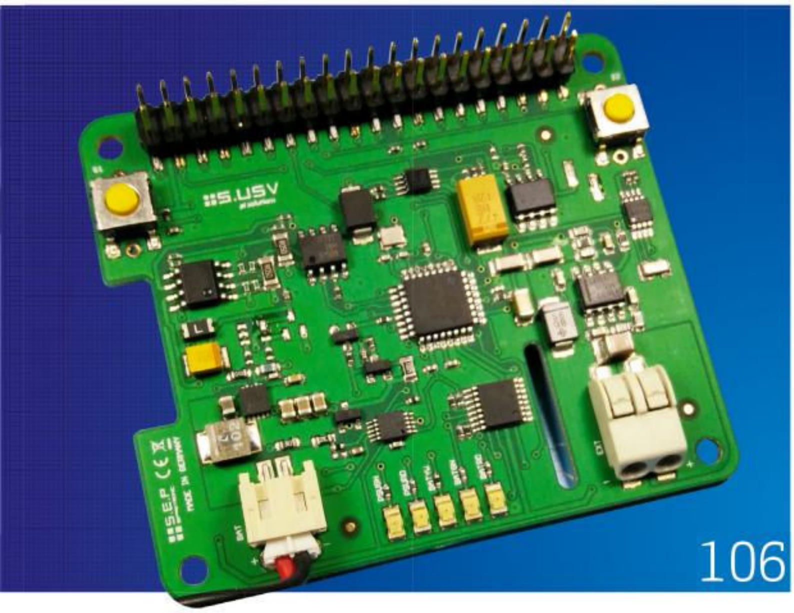
53 Make any printer wireless

56 Create a Pi-based Media Centre

61 TalkiePi: Make a walkie-talkie

66 Protect yourself with a Pi-based firewall

71 Build a Pi-hosted photo gallery



GET STARTED

New to the Pi? We help you get up and running

76 Get started with a Raspberry Pi

80 Install NOOBS with any OS

78 How to install Raspbian

82 Using networks on the Pi

Pi CODE

Learn how to code in your language of choice

86 Design a game in Scratch 2

95 Build your own Minecraft city

92 Get started with Python coding

Pi REVIEWS

Tested and rated: the latest add-ons for your Pi

102 Kano Screen Kit

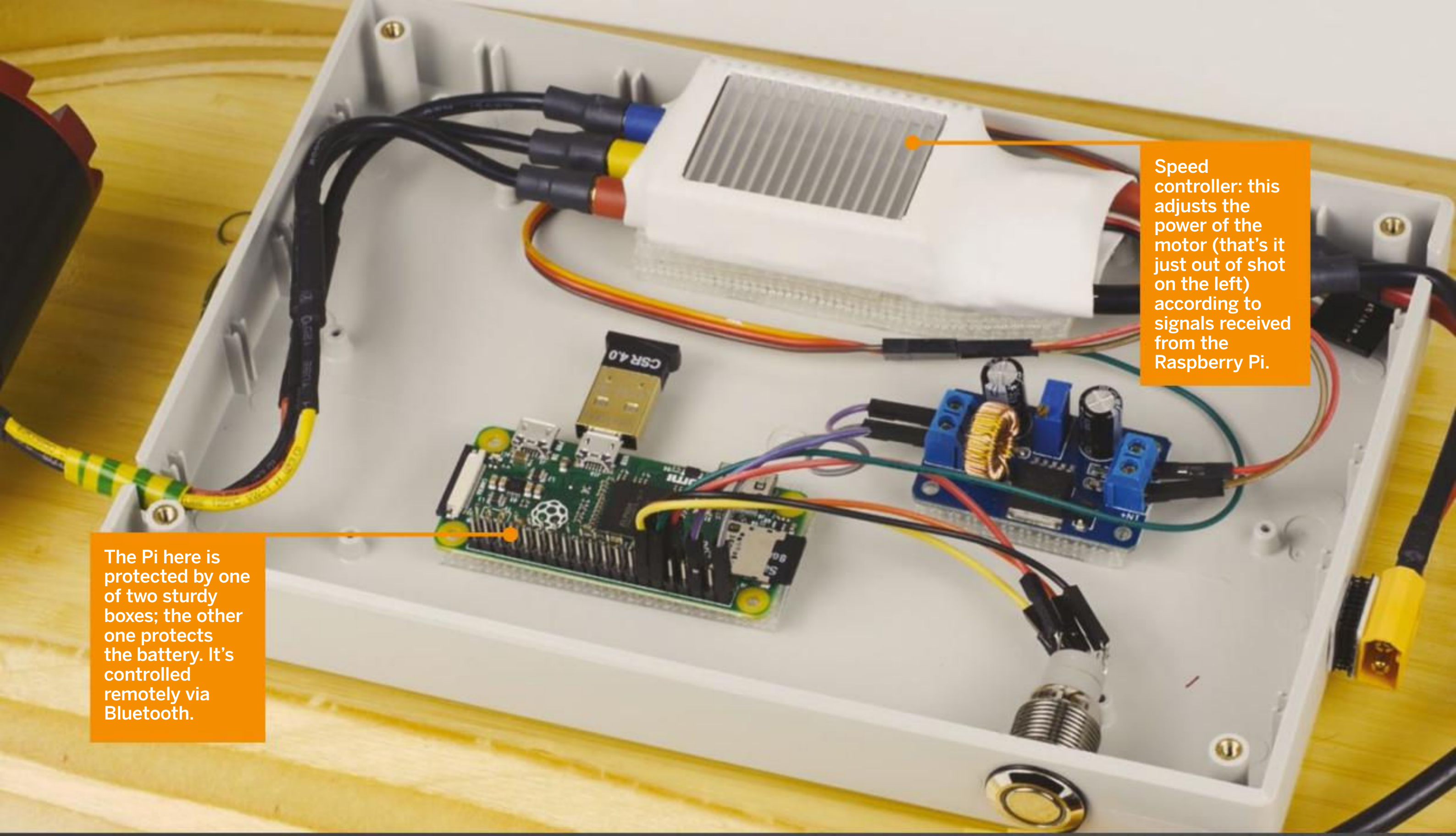
104 Pi-Top laptop kit

106 S.USV Pi Advanced

108 Enviro pHAT

110 Pi Zero RJ45 & USB Hub

112 ZeroSeg LED display kit


Speed controller: this adjusts the power of the motor (that's it just out of shot on the left) according to signals received from the Raspberry Pi.

The Pi here is protected by one of two sturdy boxes; the other one protects the battery. It's controlled remotely via Bluetooth.

Pi-powered skateboard

The Raspberry Pi Guy hits the road with a Pi-powered electric skateboard. As Pi projects go, we have to say this one's wheely good

RECIPE

- **1 x Raspberry Pi Zero**
- **1 x Nintendo Wii remote**
- **1 x brushless electric motor**
- **1 x battery**

› See the board in action at www.theraspberrypiguy.com.

Matthew Timmons-Brown is well known in Pi circles: he's the man also known as The Raspberry Pi Guy, and his site and YouTube channel feature all kinds of experiments served up to an audience of more than 57,000 people. He'll happily show you how to control motors or rig up a camera, but our favourite video shows his skateboard, which features the unlikely combination of a board, a Raspberry Pi Zero and the remote controller from a Nintendo Wii.

The skateboard is capable of an impressive 30kmh (18mph) under electric power, which "feels like a lot more when you're actually riding it," Matthew says. He hasn't tested the range too much, but says that the board is quite capable of doing ten kilometres without running out of power. It's worth pointing out that those kilometres are in Cambridge, which is very flat – we suspect that the hills of Yorkshire or San Francisco might prove to be a much tougher challenge for the board's battery (and probably the rider's stamina).

The electric skateboard's power comes from a single 3200W brushless motor that drives the rear wheels. Two extra-sturdy boxes mounted underneath the board protect the battery and the electronics from the elements, debris, accidents and the occasional hedgehog. The motor is operated by a speed controller, and that controller responds to signals from the Raspberry Pi.

Using a Wii remote is a particularly clever touch: because it's Bluetooth, the remote can connect to the skateboard's Raspberry Pi without any messy wiring, so Matthew can use it to adjust the speed of the motor without worrying about getting anything tangled. The only downside is that without the remote, the skateboard is entirely people-powered. As you'd expect, the board is steered the old-fashioned way, by leaning.

Matthew's code is around 100 lines of Python, and he's put the code up on Github so others can use it or tweak it. Sadly the skateboard itself is a one-off: Matthew's already on to his next projects, one of which is an attempt to create a Mars rover – presumably powered by a Pi. How's that for ambition?

CREATOR

MATTHEW TIMMONS-BROWN

Matthew is a 17-year-old student from Cambridge. Online, he's known as The Raspberry Pi Guy.

Chessbord

No, it's not a typo: the Chessbord is a chess board for the Internet of Things

Chess and technology go back a long way: the Mechanical Turk was a robotic – and sadly, fake – chess player that wowed crowds in the late 18th century, and chess-playing computers have been kicking around since the advent of cheap electronics in the late 1970s. And of course there's Deep Blue, IBM's chess-playing supercomputer... not to mention Ben James' Chessbord.

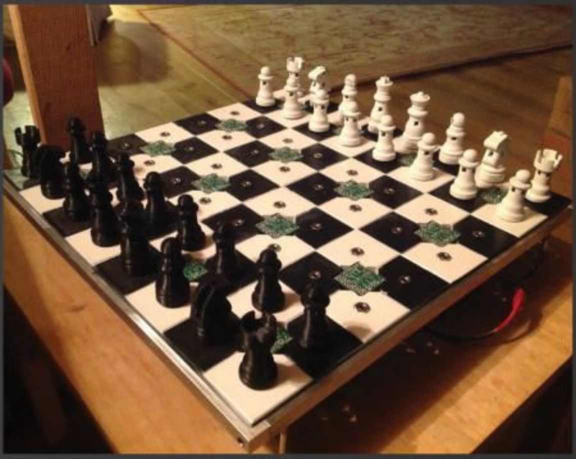
The Chessbord might not be quite as powerful (or expensive) as Deep Blue, but it's pretty clever nonetheless. Ben's design addresses a very simple problem: he likes to play chess online with others, but doesn't like playing the game on a screen. "The 2D view on the screen feels very unintuitive to me," he says. "I much prefer playing with a physical chess board."

Enter the Chessbord. "It connects to an online game so that the user can play chess with anybody in the world, with a real board," Ben explains. After initially considering some kind of robotic device to move the other player's pieces, Ben decided to go for something simpler: LEDs to show you what to move where, and sensors to detect your move and relay it to the other player. Each piece would be fitted with a jack plug, and each square would have a corresponding socket with four poles: two to power the LED and two to detect whether a piece had been inserted. Ben then connected the sockets to an Arduino, which in turn connects to a Raspberry Pi web server. The Pi communicates with the faraway player and also checks players' moves to ensure that nobody breaks any of the rules.

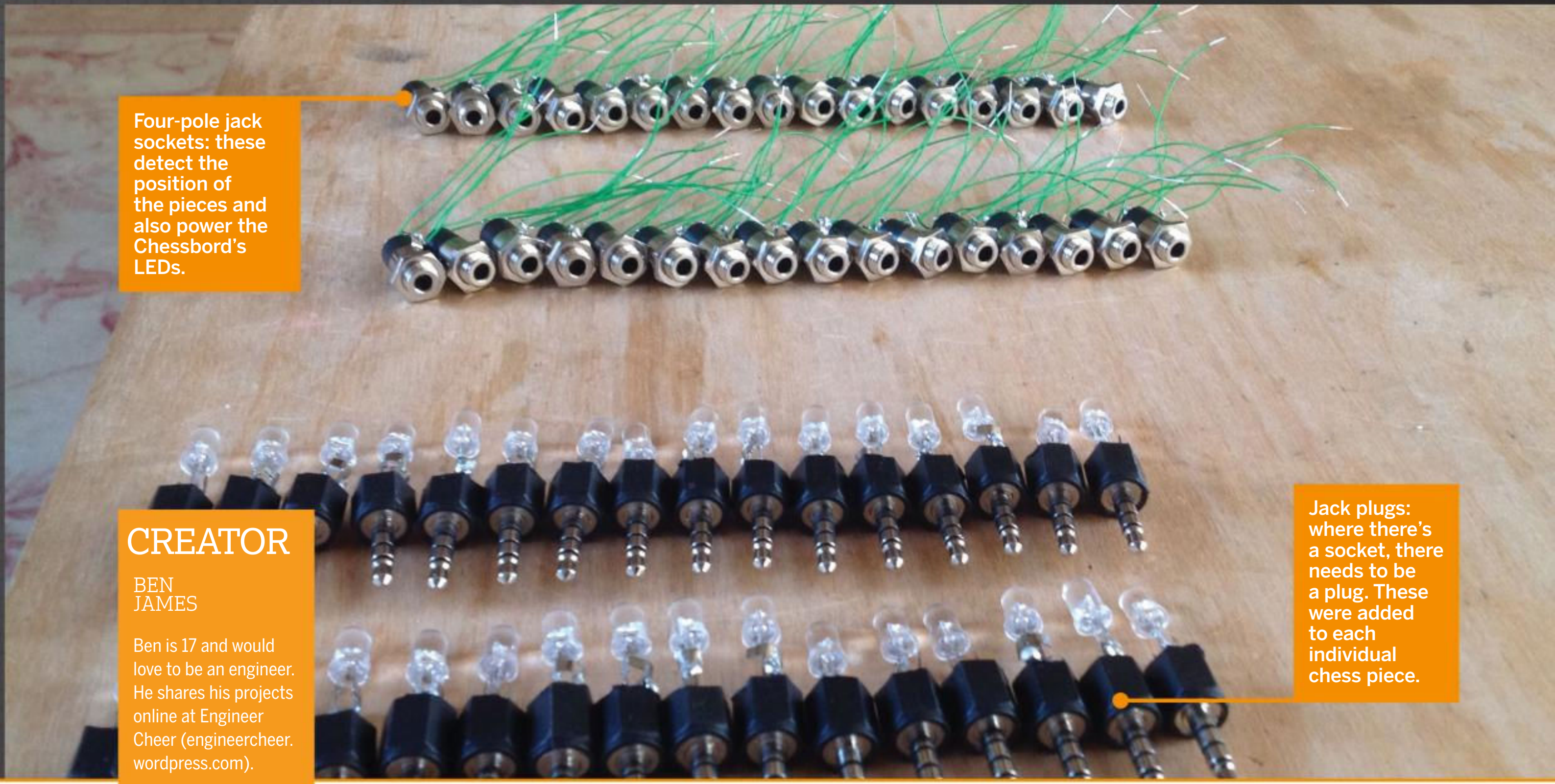
"I've always been fascinated by how things work and how they are built, and in recent years I've developed this interest into a love of electronics and coding," Ben says. "Since I was young I've been building contraptions of all shapes and sizes... A lot of my projects have micro-controllers at their heart." Ben hopes to turn his love of building things into a career in engineering. We think he's already made a really impressive start.

RECIPE

- 1 x Arduino
- 1 x Raspberry Pi
- 1 x chess board
- 64 x jack plugs, LEDs and sockets



LEDs show you where to move; sensors detect when you do so.



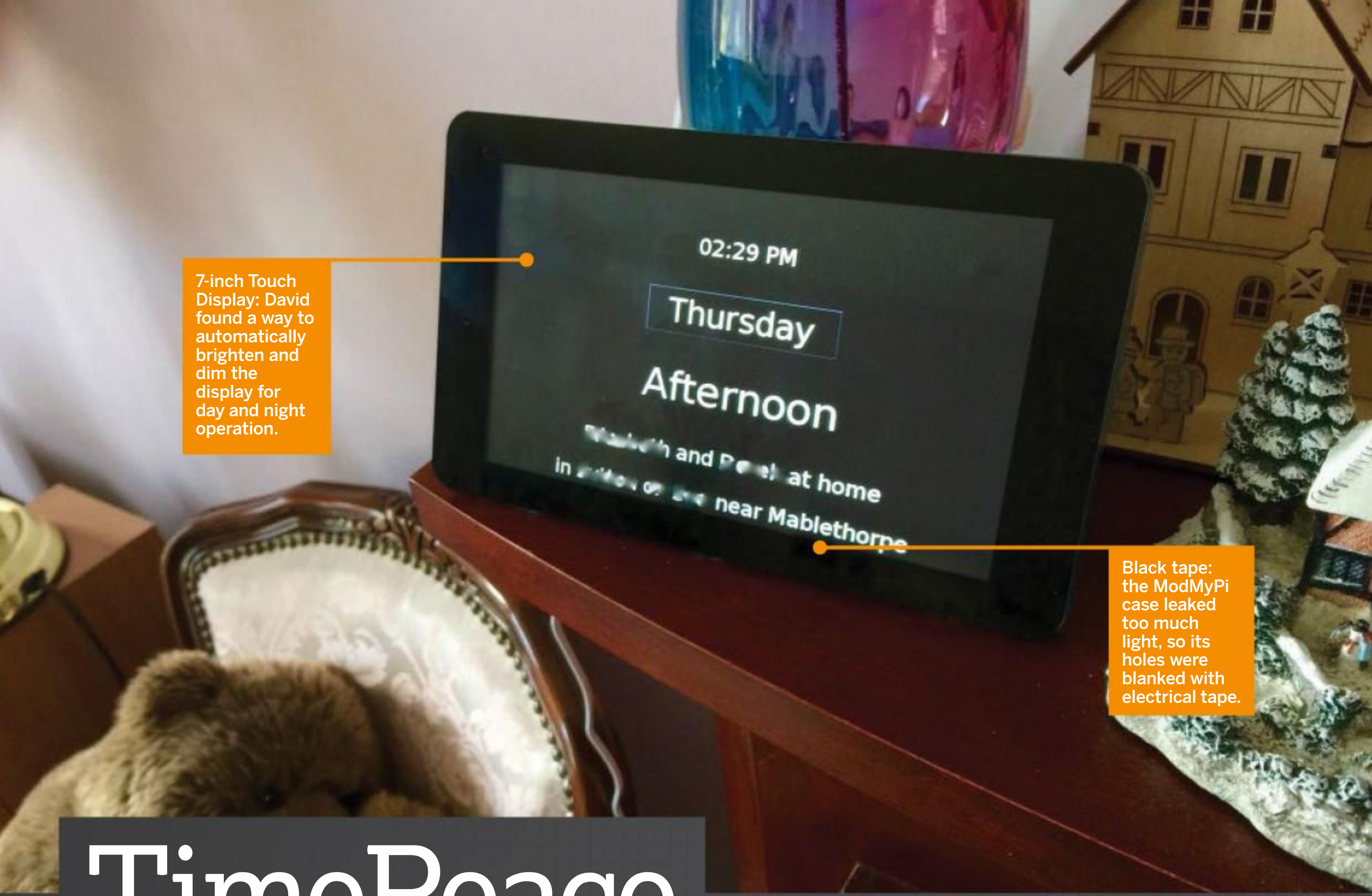
Four-pole jack sockets: these detect the position of the pieces and also power the Chessbord's LEDs.

Jack plugs: where there's a socket, there needs to be a plug. These were added to each individual chess piece.

CREATOR

BEN JAMES

Ben is 17 and would love to be an engineer. He shares his projects online at Engineer Cheer (engineercheer.wordpress.com).


02:29 PM
Thursday
Afternoon
[redacted] and [redacted] at home
in [redacted] near Mablethorpe

7-inch Touch Display: David found a way to automatically brighten and dim the display for day and night operation.

Black tape: the ModMyPi case leaked too much light, so its holes were blanked with electrical tape.

TimePeace

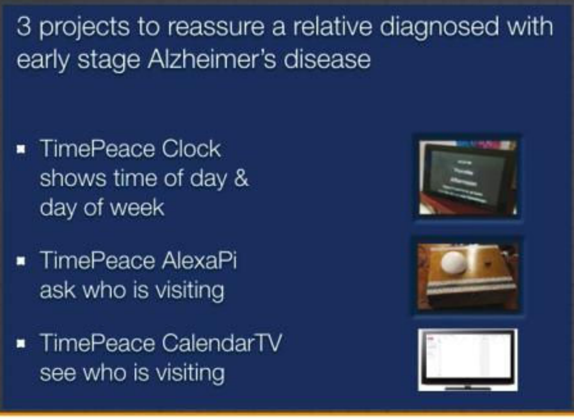
Sometimes technology can be truly life-enhancing. TimePeace is a great idea, extensively researched and thought through, and executed brilliantly

RECIPE

- ● 1 x Raspberry Pi
- ● 1 x 7in Touch Display
- ● 1 x ModMyPi case
- ● 1 x roll of black electrical tape

In 2015, David Penney's father told him that his stepmother had been diagnosed with Alzheimer's. David went to visit, and one of her comments resonated. "Her sense of time doesn't work properly," he explains. "She asks many times a day what day of the week it is." She also woke in the night wondering whether it was time to get up, and often felt that she was supposed to see someone or go somewhere without being able to recall who or where. David had a flash of insight: he could build a special clock to help her.

David undertook extensive research to identify the best way to display the information, and then he embarked on his build. "I wanted it to be an appliance, not to look like a science project, so no complex buttons and menus etc, although these are found on most clock products," he says. "So I used the Raspberry Pi Foundation's 7-inch touch display. I also wanted a robust enclosure that hides the electronics. The ModMyPi case for the display and Pi is OK but I would still prefer something that seals firmly and has an option to hide all the sockets. I used black electrical tape over all the holes, which also stops the blinking lights from the Pi from leaking out and keeping a person awake at night!"


3 projects to reassure a relative diagnosed with early stage Alzheimer's disease

- TimePeace Clock shows time of day & day of week
- TimePeace AlexaPi ask who is visiting
- TimePeace CalendarTV see who is visiting

David also had to hack the display. "The 7-inch display is also capable of dimming, but it isn't documented how to make it work," he says. "So with clues from the Pi community, I managed to reliably dim and brighten the display in software, then added a timer to tell it when to go dim at night and when to brighten in the morning."

David made two TimePeaces for his stepmother, and the results have been very positive. "The first night she woke, read 'middle of the night' on the clock and went back to sleep," he says.

David has now built three devices, taking advantage of Amazon's Alexa Voice Services for voice control, and is working on additional features in software to help his stepmother further.

CREATOR

DAVID PENNEY

David is a former IT director experienced in financial services and tech, interested in cyber defence.

› **Smart clocks** and other devices to tell you more useful stuff than just the time.

Gabian

Is it a bird? Is it a plane? No, it's a flying machine – for fishing! It will help you find where the fish are biting, drop bait and hook them for you, all remotely

Guillaume Ramirez works with helicopters and loves fishing, so it was perhaps inevitable that he'd find a way to combine the two. He's the creator of Gabian, a very ingenious fishing drone, which you can see in action at **http://gabian-en.strikingly.com**.

Gabian is a waterproof quadcopter built around a Raspberry Pi, a Navio 2 board, an RPI Cam V2 and a home-made gimbal – a device that keeps the camera steady – to stream HD video and drop fishing lines and/or bait.

"My inspiration came from buying my first Pi at the beginning of 2016 and finding out what cool things I could do with it," Guillaume says. "I'm not an IT guy, but I'm keen on fishing. I have seen some inspiring YouTube videos where people fish with their drones, so I thought, why not me?"

Unfortunately for Guillaume, "after several successful attempts my Raspberry Pi drone fell into water – which is something that happens if you try flying something very fast over water. So I wanted something waterproof that can resist crashes and even be operated from a boat without stress." He picked the name Gabian, which is the Mediterranean French name for a seagull.

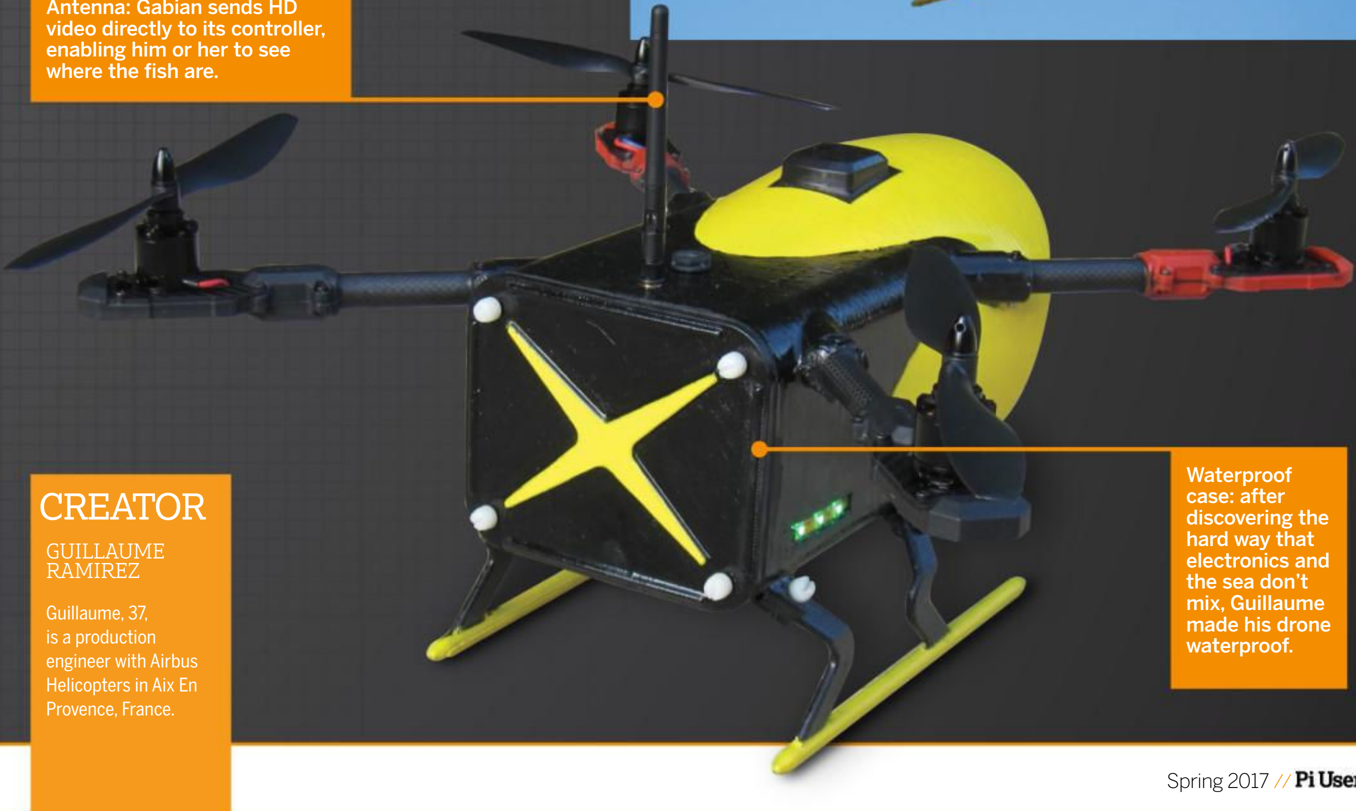
Gabian uses GPS to find the right spots, streaming live video so that the person controlling the drone can see where the fish actually are. Once they're spotted Gabian can then make repeated runs to drop bait where the fish will bite, moving the line until it hooks a fish and then handing control over to the fisherman or woman.

As you've probably guessed, Gabian is no hastily-assembled one-off: Guillaume hopes to market a new version of Gabian to switched-on fishing fanatics. "I am doing a new prototype to possibly change the manufacturing technique from 3D printing to plastic moulding," he says. "It isn't production ready yet but my target is the second quarter of 2017. I'm keeping my ambitions modest: a DIY kit would probably be the right thing to do if people are attracted by the concept." You can follow his progress and find out more about Gabian at **www.gabian-drone.com**.

RECIPE

- **2 x Raspberry Pis (the first one drowned)**
- **1 x Quadcopter**
- **1 x Navio 2 board**
- **1 x RPI Cam V2**

Fishing isn't often strenuous, but this way you don't even have to be present...



Antenna: Gabian sends HD video directly to its controller, enabling him or her to see where the fish are.

Waterproof case: after discovering the hard way that electronics and the sea don't mix, Guillaume made his drone waterproof.

CREATOR

GUILLAUME RAMIREZ

Guillaume, 37, is a production engineer with Airbus Helicopters in Aix En Provence, France.

Get the UK's best-selling
Linux magazine

FREE DVD!
OPENSUSE LEAP 42.2
LINUX FORMAT
Get into Linux today!
67 Pages of tutorials and features
» Build secret Tor websites
» Better remote desktops
» Get started with Ansible
Coding Academy: Text in Python, visual data in D3.js
YOUR OPEN SOURCE
SMART HOME
» BUILD IT!
» SECURE IT!
» CONTROL IT!
» AUTOMATE IT!
INSTALL: BOOT IT › TRY IT › LOVE IT!
64-bit
openSUSE 42.2
The enterprise-class desktop distro
NEW RELEASE
» 1,000s of apps included
» KDE Plasma 5 desktop
» Next-gen Btrfs storage
» Super stable build
TRY IT TODAY!
LINUX KERNEL 4.4
STABLE 'LTS' RELEASE
SNAPPER BACKUP
TIME MACHINE
SERVER RELEASE
CLOUD READY
LINUX FORMAT DVD
BOOT DISC: INSTALL NOW
EVERYTHING YOU NEED TO GET STARTED WITH LINUX
+ Raspberry Pi tips & tricks
» Turn Minecraft open source
» Building menus with Whiptail
» Raspberry Pi 2 v1.2 revealed!
Future
OUT NOW!

DELIVERED DIRECT TO YOUR DOOR

Order online at **www.myfavouritemagazines.co.uk**

or find us in your nearest supermarket, newsagent or bookstore!

THE TOP 20 Raspberry Pi apps

Discover how to do even more with your Raspberry Pi as **Nate Drake** reveals our pick of the very best applications available for the Pi today

The Raspberry Pi is probably best known as a platform for programming and physical computing projects, but it's also a fully-functional computer, and like desktop computers and modern mobile devices of all kinds, it has a wide range of apps available to expand its capabilities. What you install will of course depend on what you want to do with your Pi, but this is our pick of 20 apps that we love, and that all work on the Pi with a default Raspbian install.

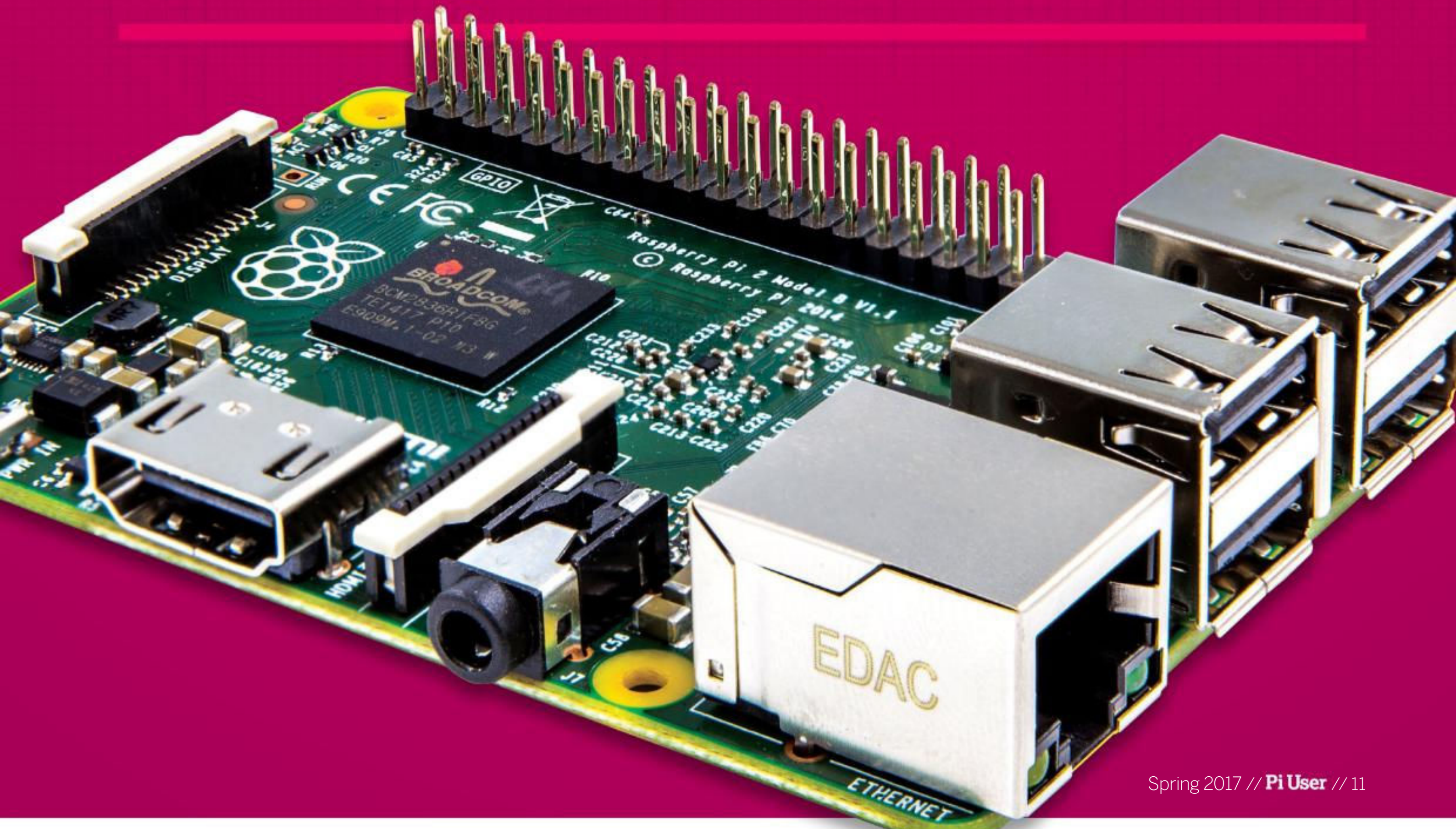
While the Raspberry Pi no longer has an online "App Store" – the Pi Store ceased operating in early 2015 – some handy programs come pre-installed and many more can be installed without much trouble. Click Applications > Preferences > Add/Remove Software to launch Raspbian's Package Manager. You'll see that the available programs have been neatly categorised to make it easy to sort through them. There's even a small description of individual programs at the bottom to help you decide what to install. This is especially helpful if you haven't used Linux before and need to find an app equivalent to one you might have used in Windows. For instance LibreOffice's Writer is an excellent substitute for Microsoft Office.

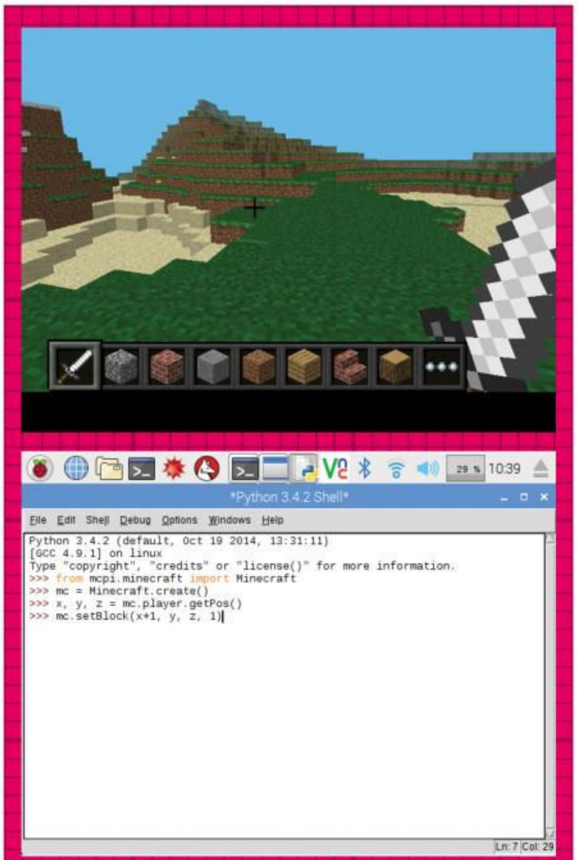
In order to find any of the programs listed over these pages, click the search box at the top left of the Add/Remove Software window, enter your search terms and press Return. Click the tickbox next to the name of individual apps you want to add, then click Apply at the bottom right in order to install them. Some of the programs listed here will already be installed on your Pi, in which case they will already be ticked.

Even if your Pi is not connected to a monitor, you can install programs by connecting to it via SSH. You will first need the name of the program. You can then install it using **sudo apt-get**, for example as follows:

```
sudo apt-get dosbox
```

Read on for the most exciting apps on the Pi today...! »





Python 3.4.2 Shell
File Edit Shell Debug Options Windows Help
10:39
Ln: 7 Col: 29

```
Python 3.4.2 (default, Oct 19 2014, 13:31:11)
[GCC 4.9.1] on linux
Type "copyright", "credits" or "license()" for more information.
>>> from mcpi.minecraft import Minecraft
>>> mc = Minecraft.create()
>>> x, y, z = mc.player.getPos()
>>> mc.setBlock(x+1, y, z, 1)
```

Play the game, and make your first foray into Python programming by building Minecraft blocks.

1

Minecraft Pi

Play the popular building game for free

Minecraft Pi is a version of the insanely successful Minecraft game written specifically for the Raspberry Pi. It is based on the Pocket edition, so may not have all the features of the full-blown version. Gameplay is "Creative" mode only, and multiplayer with users of the regular Minecraft game is not supported. The advantage to this, however, is that no one can destroy your structures.

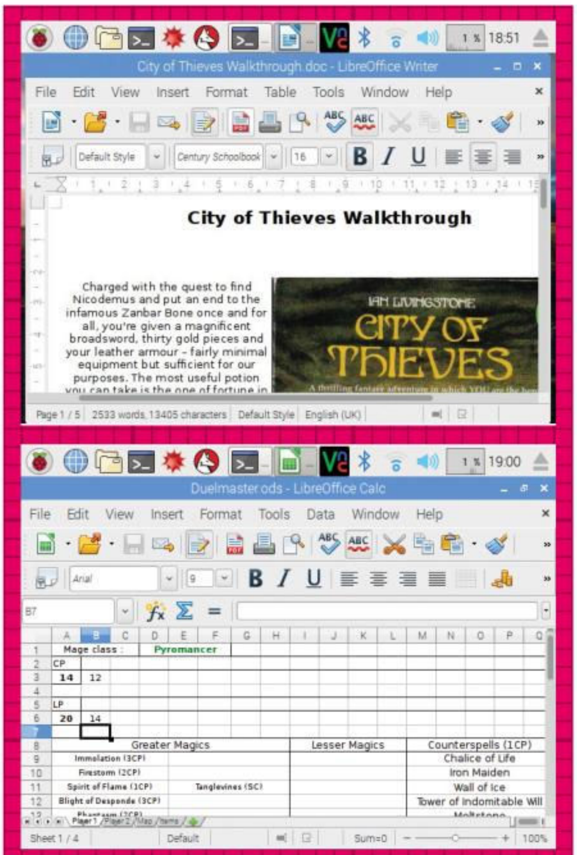
If you want to get more from Minecraft or just try Python programming, press the Tab key and open Python 3 (IDLE) from Applications > Programming. From here you can type commands to automatically drop blocks as you walk, automatically create structures, and teleport yourself from place to place. For instance, to place a stone block immediately behind you, type:

```
x, y, z = mc.player.getPos()
mc.setBlock(x+1, y, z, 1)
```

Visit the official blog at **http://bit.ly/minecraft-pi-guide** for a worksheet of basic commands.

You can find Minecraft Pi pre-installed in Applications > Games on your Pi.

The Raspberry Pi will also happily run a Minecraft server, allowing you and your friends to build your own virtual world provided you each have the official version of the game. There are a number of guides available to do this. We recommend that you use the SpigotMC server software as it runs well on the Pi.



City of Thieves Walkthrough.doc - LibreOffice Writer
File Edit View Insert Format Table Tools Window Help
Default Style
Century Schoolbook
16
18:51
IAN LIVINGSTONE
CITY OF THIEVES
Page 1 / 5
2533 words, 13405 characters
Default Style
English (UK)
Duelmaster.ods - LibreOffice Calc
File Edit View Insert Format Tools Data Window Help
Arial
9
B7
19:00
Player 1 / Player 2 / Map / Items
Sheet 1 / 4
Default
Sum=0
100%

City of Thieves Walkthrough

Charged with the quest to find Nicodemus and put an end to the infamous Zanbar Bone once and for all, you're given a magnificent broadsword, thirty gold pieces and your leather armour – fairly minimal equipment but sufficient for our purposes. The most useful potion you can take is the one of fortune in

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Mage class :			Pyromancer									
2	CP												
3	14	12											
4													
5	LP												
6	20	14											
7													
8	Greater Magics								Lesser Magics				Counterspells (1CP)
9	Immolation (3CP)												Chalice of Life
10	Firestorm (2CP)												Iron Maiden
11	Spirit of Flame (1CP)			Tanglevines (SC)									Wall of Ice
12	Blight of Desponde (3CP)												Tower of Indomitable Will

The LibreOffice interface will be very intuitive to anyone who has used office software before.

2

LibreOffice

Find all the office apps you need

Want to use office apps on your Pi, for free? LibreOffice provides a familiar looking word processor (Writer), spreadsheet software (Calc), presentation program (Impress) and even an app for creating databases (Base).

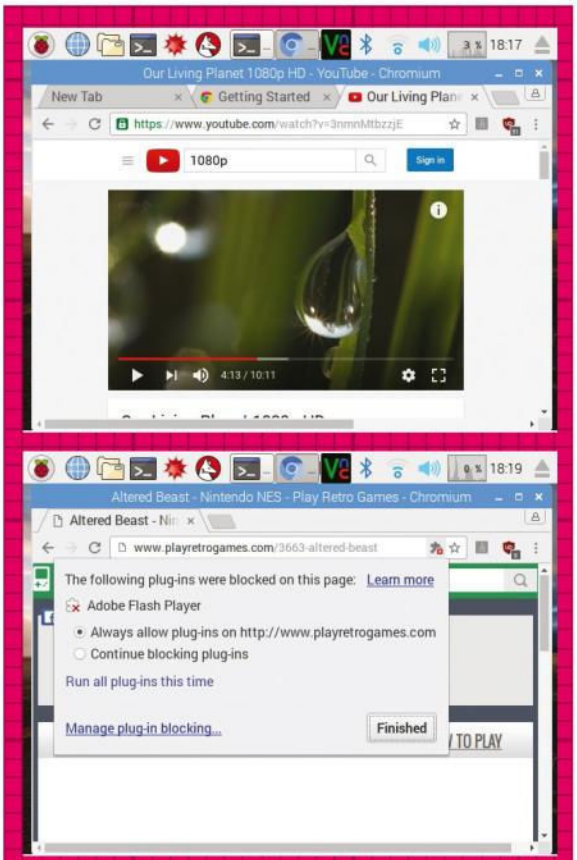
Need to share files with Mac or Windows users? LibreOffice bridges the gap: it supports ODF (Open Document Format) by default but can also open files created in Microsoft Word, Excel and Powerpoint, as well as save files in Microsoft-compatible .docx, .xlsx and .pptx formats (among many others).

Tired of Americanisms such as "color" appearing in your work? Unlike Microsoft Office, LibreOffice can install a version that contains only UK English.

Another extremely useful feature is that with one click you can convert documents to PDF format – useful because PDFs generally appear exactly the same on every computer.

Help for specific programs and features is available at the LibreOffice Community Support page at **www.libreoffice.org/get-help/community-support**. If you like the app on the Pi, you can also download LibreOffice for Mac or Windows from **http://libreoffice.org**.

The LibreOffice apps Base, Calc, Draw, Impress, Math and Writer come pre-installed on the Pi. Note that the Draw program can run slowly on older versions of the Pi – use the Raspberry Pi 3 for best performance.


18:17
Our Living Planet 1080p HD - YouTube - Chromium
New Tab
Getting Started
Our Living Plan
https://www.youtube.com/watch?v=3nmnMtbzzjE
1080p
Sign in
4:13 / 10:11
18:19
Altered Beast - Nintendo NES - Play Retro Games - Chromium
Altered Beast - Nin
www.playretrogames.com/3663-altered-beast
Learn more
Adobe Flash Player
Run all plug-ins this time
Manage plug-in blocking...
Finished
TO PLAY

The following plug-ins were blocked on this page:

● Always allow plug-ins on http://www.playretrogames.com
○ Continue blocking plug-ins

By default the Adobe Flash plugin is disabled. Click "Always allow plugins" to enable them site by site.

3

Chromium

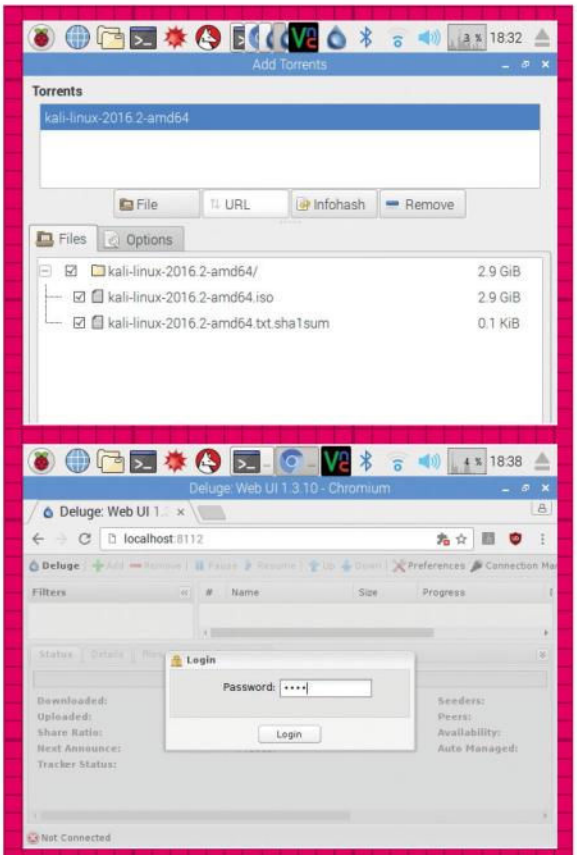
Surf the web in style with a shiny browser

Chromium replaced the former web browser Epiphany in the last major update for the Pi. As the similarity of the name hints, it's related to Google's Chrome web browser – Chrome derives most of its source code from the open source Chromium Project. One consequence of this is that you can install many Chrome addons such as the Google Hangouts plugin from the Chrome Web Store (**https://chrome.google.com/webstore/category/extensions?**). Chromium will support most of your internet needs, although you may find that playing certain online games or streaming high quality video content is sluggish. To assist with this, the good people at the Raspberry Pi Foundation have included the h264ify browser extension, which makes YouTube display videos in a format that can be accelerated by the Pi's hardware.

Also bundled with Chromium is the uBlock Origin extension, which blocks most annoying popups and ads. They're not just a pain but can slow down the Pi dramatically if left unchecked.

If you do not already have Chromium installed on your Raspberry Pi, back up any personal data, then open the Terminal or connect via SSH and run the following commands, one after the other, to upgrade to the latest version of Raspbian complete with this new browser:

```
sudo apt-get update
sudo apt-get upgrade
```


18:32
Add Torrents
Torrents
File
URL
Infohash
Remove
Files
Options
18:38
Deluge: Web UI 1.3.10 - Chromium
Deluge: Web UI 1...
localhost:8112
Deluge
Add
Remove
Pause
Resume
Up
Down
Preferences
Connection Manager
Filters
Name
Size
Progress
Status
Details
Downloaded:
Uploaded:
Share Ratio:
Next Announce:
Tracker Status:
Seeders:
Peers:
Availability:
Auto Managed:
Login
Not Connected

kali-linux-2016.2-amd64

☑ kali-linux-2016.2-amd64/ 2.9 GiB
☑ kali-linux-2016.2-amd64.iso 2.9 GiB
☑ kali-linux-2016.2-amd64.txt.sha1sum 0.1 KiB

Password: ••••

Access the Deluge web interface from another device by browsing to http://ipaddressofyourpi:8112.

4

Deluge

Download files via BitTorrent

The Pi supports a number of apps for downloading via BitTorrent, the peer-to-peer file sharing protocol, but Deluge trumps them all in terms of the sheer number of features.

If you run your Pi without a monitor, Deluge can be accessed via a web interface from another device. This is extremely useful if you wish to download files while away from home. Simply open a browser and go to **http://ipaddressofyourpi:8112**. Deluge is protected by a password (the default is "deluge"). Instructions are on the Deluge Wiki at **http://dev.deluge-torrent.org/wiki/UserGuide/ThinClient**.

While there is nothing illegal in itself about downloading files via BitTorrent (most Linux distributions can be downloaded this way), your ISP may restrict or throttle BitTorrent traffic. Consider using Deluge's Blocklist plugin to automatically block IP addresses of organisations who are likely to try to prevent you from using BitTorrent. See **http://dev.deluge-torrent.org/wiki/Plugins/Blocklist** for full instructions. Blocklists themselves can be found at **www.iblocklist.com/lists.php**.

You can install the Deluge program using the Add/Remove Software window. If you're running your Pi without a monitor and want to control Deluge via SSH run the following:

```
sudo apt-get install deluged
deluge-console python-mako
deluge-web
```

»

5

VLC Player

Play and stream any type of media

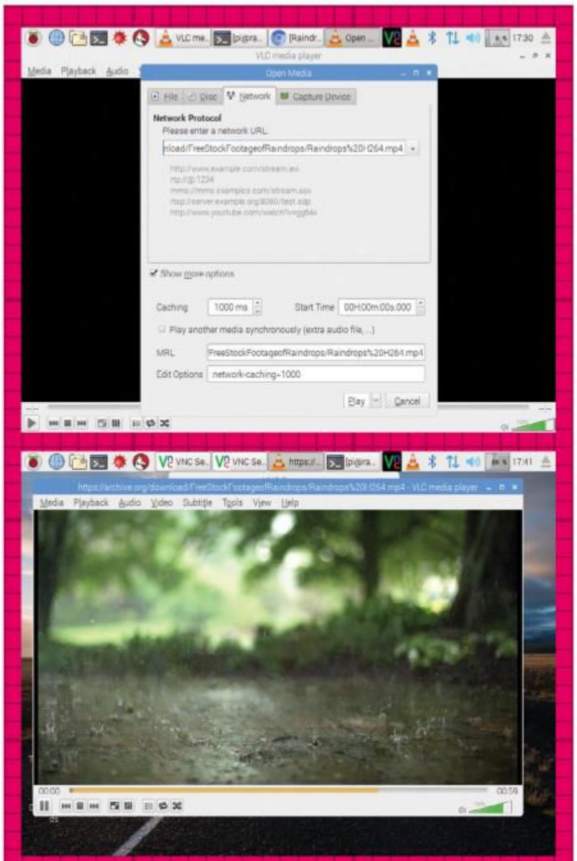
To put it quite simply, VLC plays everything, from MP3 files, to Apple's QuickTime format, to Windows Media Videos. This is especially useful on the Pi as by default the Pi cannot play some of the more exotic and non-free media formats. VLC also has a handy plugin for your web browser to allow playing of media files from websites.

Although it is best known for being a universal media player with an unassuming interface, VLC has some extraordinarily powerful extra features. For instance if you wish to record yourself using the Pi, VLC's Capture Device option can record whatever is running on your desktop. VLC can also download online videos such as YouTube clips from its Network tab.

Another brilliant feature is the ability to convert files from one audio/video format to another. This is especially useful for people with Smart TVs that can play only .avi video files.

Speaking of TVs, a Raspberry Pi running VLC and plugged into a TV can become a handy media centre, controlled by a web browser. Enable this option in VLC's preferences to allow you to select, pause and play music or videos from another device.

Install VLC player via Add/Remove Software. In order to play the range of video files supported by VLC the system may ask you to agree to install some additional programs.


VLC media player
Open Media
File
Disc
Network
Capture Device
Network Protocol
Please enter a network URL:
Show more options
Play
Cancel
Media
Playback
Audio
Video
Subtitle
Tools
View
Help

nload/FreeStockFootageofRaindrops/Raindrops%20H264.mp4

Caching: 1000 ms

Start Time: 00H:00m:00s.000

MRL: FreeStockFootageofRaindrops/Raindrops%20H264.mp4

Edit Options: :network-caching=1000

VLC will play almost any media, and on a Pi connected to a TV can be the heart of a powerful media centre.

6

IceDove

Let your emails soar

IceDove is a version of the popular Mozilla Thunderbird email program. Although the Pi does come with a built-in email client (Claws Mail), IceDove offers much more customisation.

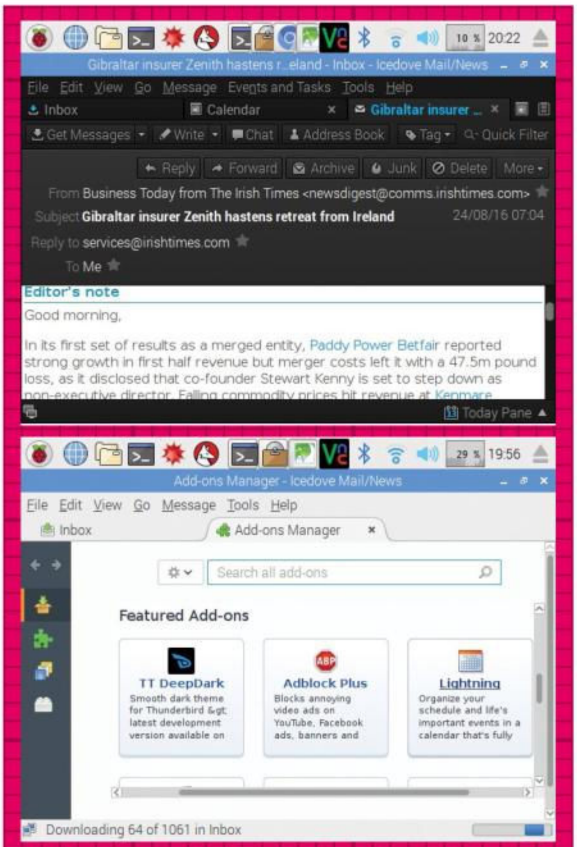
The mail setup wizard is very easy to follow, and all common providers' settings can be found in the Mozilla database. In most cases you will simply need to enter your email address and password, and IceDove will then do the rest.

Aside from managing your emails, it can also manage your news feeds and supports several chat services such as Google Hangouts. IceDove also supports Filelink: as an alternative to sending large files, you have the option to store attachments in the cloud and send your recipient a link to them.

You can extend IceDove even further by heading to the Add/Remove Software window and installing iceowl, which will install a fully-featured calendar; another extension will also add support for Google Calendars.

The default interface looks rather dry but you can change this by installing one of the many colourful themes. Go to Tools > Add Ons to see some of those on offer. There are also software add-ons for IceDove such as Enigmail, which allows you to encrypt your emails.

Install IceDove by going to Preferences > Add/Remove Software and entering the name in the search bar.


Gibraltar insurer Zenith hastens r...eland - Inbox - Icedove Mail/News
File
Edit
View
Go
Message
Events and Tasks
Tools
Help
Inbox
Calendar
Gibraltar insurer ...
Get Messages
Write
Chat
Address Book
Tag
Quick Filter
Reply
Forward
Archive
Junk
Delete
More
Today Pane
Add-ons Manager - Icedove Mail/News
Add-ons Manager
Search all add-ons
Featured Add-ons
TT DeepDark
Adblock Plus
Lightning

From Business Today from The Irish Times <newsdigest@comms.irishtimes.com>

Subject Gibraltar insurer Zenith hastens retreat from Ireland 24/08/16 07:04

Reply to services@irishtimes.com

To Me

Editor's note

Good morning,

In its first set of results as a merged entity, Paddy Power Betfair reported strong growth in first half revenue but merger costs left it with a 47.5m pound loss, as it disclosed that co-founder Stewart Kenny is set to step down as non-executive director. Falling commodity prices hit revenue at Kenmare

Smooth dark theme for Thunderbird &gt; latest development version available on

Blocks annoying video ads on YouTube, Facebook ads, banners and

Organize your schedule and life's important events in a calendar that's fully

Downloading 64 of 1061 in Inbox

IceDove is a more customisable email client, with news feeds and chat, plus a range of available add-ons.

7

Synaptic Package Manager

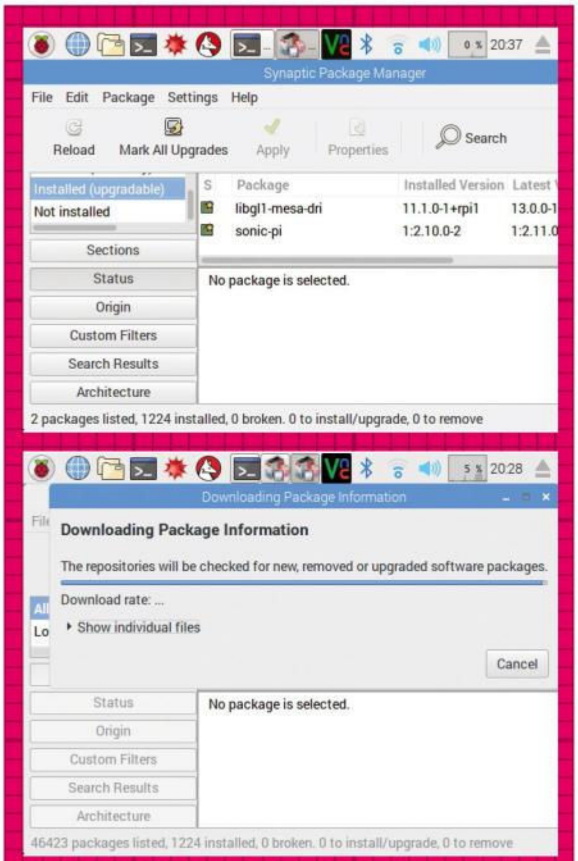
Turbocharge your package management

Like Add/Remove Software in Applications > Preferences, Synaptic is a pleasant graphical front end for installing the vast number of programs available for the Pi. But if you already have something to install new apps, why would you need Synaptic? First, Synaptic contains a good many more categories than the very broad ones offered by Add/Remove Software. Instead of one "internet" section, for instance, you can select "Communication, "Email", "Networking" and so on.

Synaptic can also ensure that you have the most up-to-date version of each program. Click the Reload button to scan the available software for Raspbian and update the Pi.

Another huge advantage of Synaptic is that if you select a program for installation that requires others in order to work properly, it will automatically mark these for download too.

The buttons at the bottom left help to narrow down the various programs much more easily too. For instance clicking the Status button followed by Installed in the window above will list all installed programs. Click Mark all Upgrades to automatically flag any programs that can be updated. When ready, click Apply to download the relevant files.


Synaptic Package Manager
File Edit Package Settings Help
Reload
Mark All Upgrades
Apply
Properties
Search
Installed (upgradable)
Not installed
Sections
Status
Origin
Custom Filters
Search Results
Architecture
20:37
Downloading Package Information
Cancel
20:28

S	Package	Installed Version	Latest
	libgl1-mesa-dri	11.1.0-1+rpi1	13.0.0-1
	sonic-pi	1:2.10.0-2	1:2.11.0

No package is selected.

2 packages listed, 1224 installed, 0 broken. 0 to install/upgrade, 0 to remove

Downloading Package Information

The repositories will be checked for new, removed or upgraded software packages.

Download rate: ...

Show individual files

No package is selected.

46423 packages listed, 1224 installed, 0 broken. 0 to install/upgrade, 0 to remove

Add/Remove Software is perfectly adequate. But Synaptic is much more, with useful extra capabilities.

8

Pidgin

Manage all your Messaging from one place

The popular cross-platform messaging app Pidgin saves you the trouble of managing multiple accounts like Google Hangouts, AOL Instant Messenger and Yahoo Messenger. All your chats can be combined in one place. More chat protocols such as Skype and WhatsApp are available via plugins (see below). Sadly since Facebook dropped support for the open XMPP standard in 2014, Pidgin can no longer be used to chat with Facebook contacts via the site.

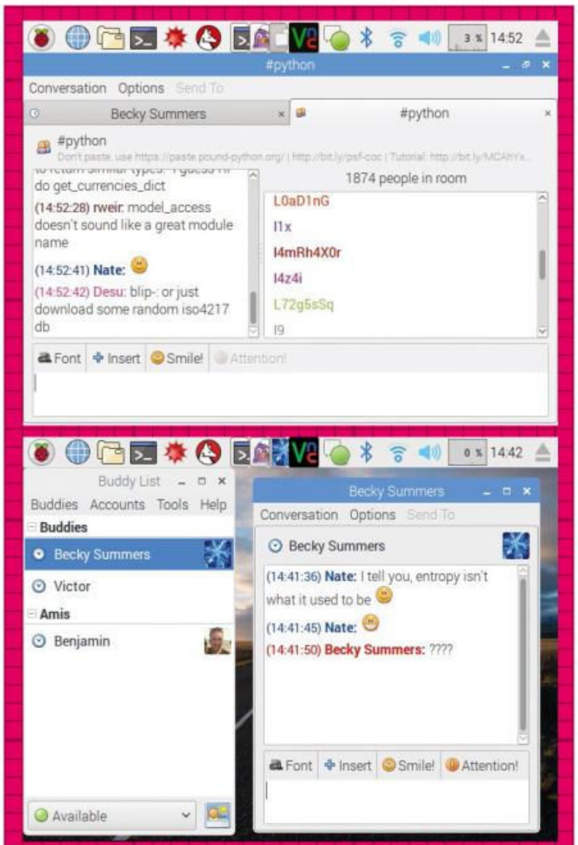
Pidgin does however also support IRC Chatrooms. By default it is configured for the huge Freenode network. Users can join "channels" to discuss a huge number of topics from programming to jam making.

Pidgin can also be configured in many ways. You can change themes (icons, emoticons, fonts etc.) by choosing from the thousands available online – see **https://developer.pidgin.im/wiki/ThemingPidgin**.

For Pidgin plugins, see **https://developer.pidgin.im/wiki/ThirdPartyPlugins**. Some of these are quite benign and will extend the messaging protocols Pidgin supports. The ultra paranoid can add the OTR (Off the Record) messaging plugin, which will encrypt conversations to allow you to chat privately over the internet.

Install Pidgin by going to Preferences > Add/Remove Software and searching for it using the bar at the top left.

»


14:52
#python
Conversation Options Send To
Becky Summers
#python
1874 people in room
L0aD1nG
I1x
I4mRh4X0r
I4z4i
L72g5sSq
I9
Font
Insert
Smile!
Attention!
14:42
Buddy List
Buddies Accounts Tools Help
Buddies
Becky Summers
Victor
Amis
Benjamin
Available
Becky Summers
Conversation Options Send To

#python

Don't paste, use https://paste.pound-python.org/ | http://bit.ly/psf-coc | Tutorial: http://bit.ly/MCAhYx...

do get_currencies_dict

(14:52:28) rweir: model_access doesn't sound like a great module name

(14:52:41) **Nate:**

(14:52:42) Desu: blip-: or just download some random iso4217 db

(14:41:36) **Nate:** I tell you, entropy isn't what it used to be

(14:41:45) **Nate:**

(14:41:50) **Becky Summers:** ????

Pidgin supports multiple protocols so you can chat to a number of friends at once.

9

GIMP

Edit all your photos professionally on the Pi

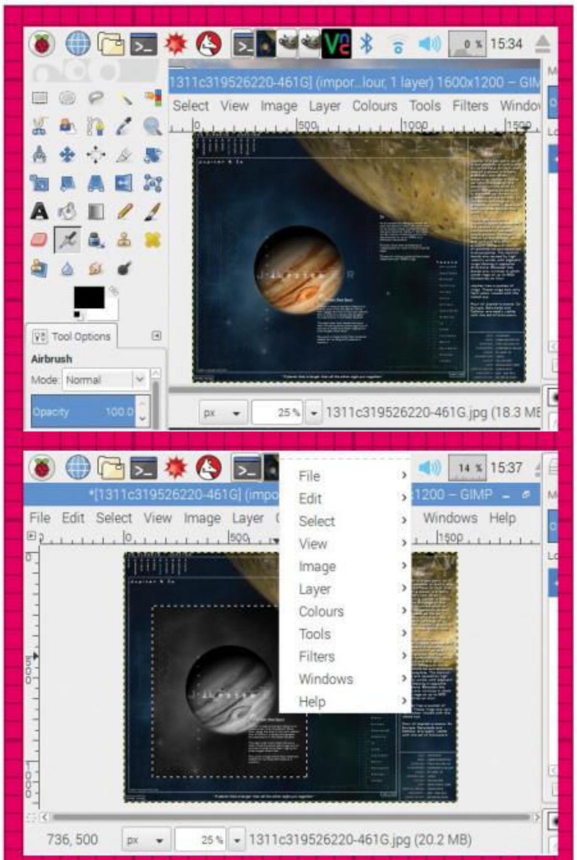
GIMP is an advanced image editor. It's designed to be a serious alternative to some of the expensive paid-for photo-editing apps while still being intuitive to use. GIMP contains all the tools you'll need for basic image editing such as altering the contrast, cropping or cutting parts of an image. GIMP can also open and convert various types of image. This is particularly handy if you have a digital camera which takes high quality photos that you wish to reduce in size to share over the web. Choose File > Export to make them compact JPGs or similar.

More advanced features such as selective colorisation are available through use of layer masks, allowing you to control the level of transparency and colour of parts of any image.

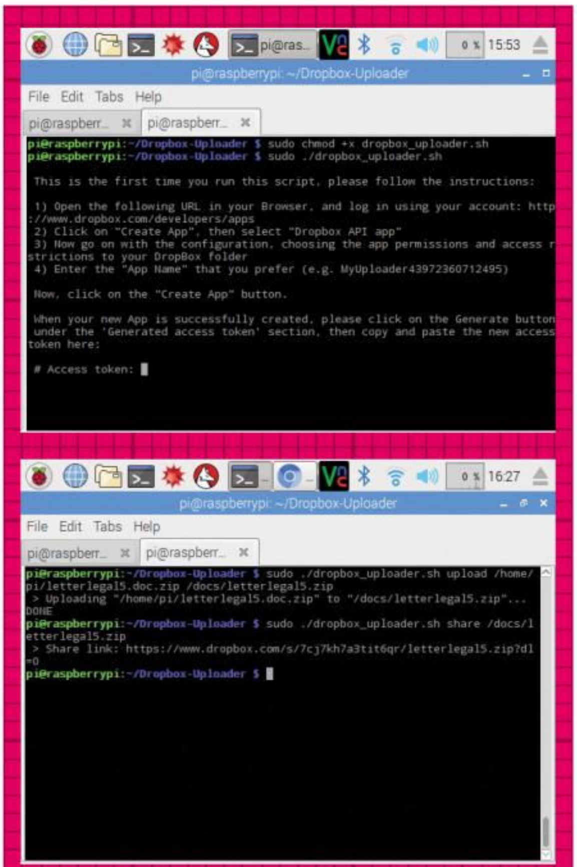
Pi users can also make use of scripts written in Perl to automate common tasks – for example to bulk-convert a load of .xcf images to .jpg and add a caption. You'll find sample Perl scripts plus plugins to extend GIMP's functions even further in the GIMP Plugins Registry: **http://registry.gimp.org**.

You might be surprised how professional your images can look after you spend just a little time with GIMP. If you want to explore further, **www.gimp.org/tutorials** offers free tutorials.

Install GIMP by clicking Preferences > Add/Remove Software and searching for it using the bar at the top left.



You can select individual parts of an image to edit the image further and/or export to a new image.



```
pi@raspberrypi:~/Dropbox-Uploader $ sudo chmod +x dropbox_uploader.sh
pi@raspberrypi:~/Dropbox-Uploader $ sudo ./dropbox_uploader.sh

 This is the first time you run this script, please follow the instructions:

 1) Open the following URL in your Browser, and log in using your account: http
://www.dropbox.com/developers/apps
 2) Click on "Create App", then select "Dropbox API app"
 3) Now go on with the configuration, choosing the app permissions and access r
strictions to your DropBox folder
 4) Enter the "App Name" that you prefer (e.g. MyUploader43972360712495)

 Now, click on the "Create App" button.

 When your new App is successfully created, please click on the Generate button
 under the 'Generated access token' section, then copy and paste the new access
token here:

 # Access token: █
```

```
pi@raspberrypi:~/Dropbox-Uploader $ sudo ./dropbox_uploader.sh upload /home/
pi/letterlegal5.doc.zip /docs/letterlegal5.zip
 > Uploading "/home/pi/letterlegal5.doc.zip" to "/docs/letterlegal5.zip"...
DONE
pi@raspberrypi:~/Dropbox-Uploader $ sudo ./dropbox_uploader.sh share /docs/l
etterlegal5.zip
 > Share link: https://www.dropbox.com/s/7cj7kh7a3tit6qr/letterlegal5.zip?dl
=0
pi@raspberrypi:~/Dropbox-Uploader $ █
```

Select "Full Access" on Dropbox Uploader's developer page to allow the script to control your account.

10

Dropbox Uploader

Sync, edit and delete your Dropbox files

There's no official Dropbox client on the Pi, although some other cloud sharing services such as Sparkleshare are supported. You can always access Dropbox by logging in using the Chromium browser, but Andrea Fabrizi has created a command line script for basic Dropbox functions.

First connect via SSH or open Terminal on your Pi and get the necessary software with:

```
git clone https://github.com/andreafabrizi/Dropbox-Uploader.git
```

Make the script executable and run it with these commands:

```
sudo chmod +x dropbox_uploader.sh
sudo ./dropbox_uploader.sh
```

You'll then need to visit **www.dropbox.com/developers** and create a new app to allow the script to interface with your Dropbox account. Give it "full access". Once the app has been created, click Generate Token. Now copy and paste this into the Terminal window. The script will save it, and then will be able to access your Dropbox.

A full list of all commands is available at **https://github.com/andreafabrizi/Dropbox-Uploader** but the basic format for uploading is as follows:

```
.dropbox_uploader.sh upload pathtolocalfile /pathtoremotefile
```

For instance:

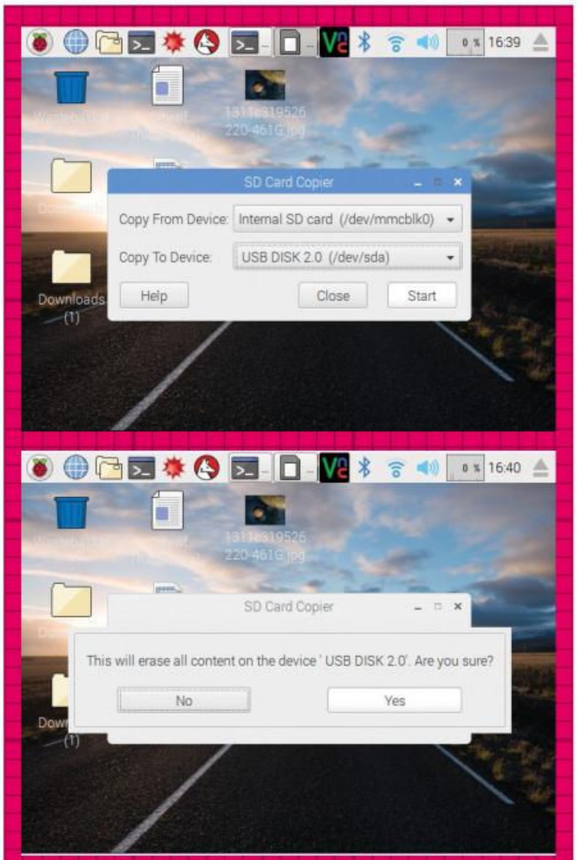
```
sudo ./dropbox_uploader.sh upload /home/pi/letterlegal5.doc.zip /docs/letterlegal5.zip
```

The script displays a progress bar as the file uploads.

11

SD Card Copier

Quickly and easily back up your entire Pi


SD Card Copier

Copy From Device: Internal SD card (/dev/mmcblk0)

Copy To Device: USB DISK 2.0 (/dev/sda)

This will erase all content on the device ' USB DISK 2.0'. Are you sure?

Help
Close
Start
No
Yes
Downloads (1)
13118319526 220-461G.jpg
16:39
16:40

It looks simple, but SD Card Copier is very powerful. Back up any data already on the target drive before use.

As you start producing photos, personal emails and intricate Minecraft structures on your Pi, you'll want a reliable way to back everything up. While there are various backup utilities available to install on the Pi, the built-in SD Card Copier will likely meet your needs. It's at Applications > Preferences > SD Card Copier.

To begin the copy process, attach an external disk or other storage device and then select it in the Copy To menu. Be warned that the destination disk will be formatted before copying – any files already on it will be lost.

After copying, the resulting drive will be an exact copy of the Pi's SD card, right down to the individual partitions. This can be problematic if you're using a larger external drive for other purposes, so consider devoting a USB stick solely to backing up/transferring data.

SD Card Copier is also very useful if you need more space for new programs. Insert a higher capacity SD card into a card reader and connect to the Pi, then begin the copy process. This will preserve all your passwords, applications and other settings.

If you have problems with data corruption or media failure, remove the current SD card and insert the backup one. You can then use the utility in reverse, copying the contents of the current backup card to the one you previously removed or restoring to a brand new card.

12

Zip Archiver

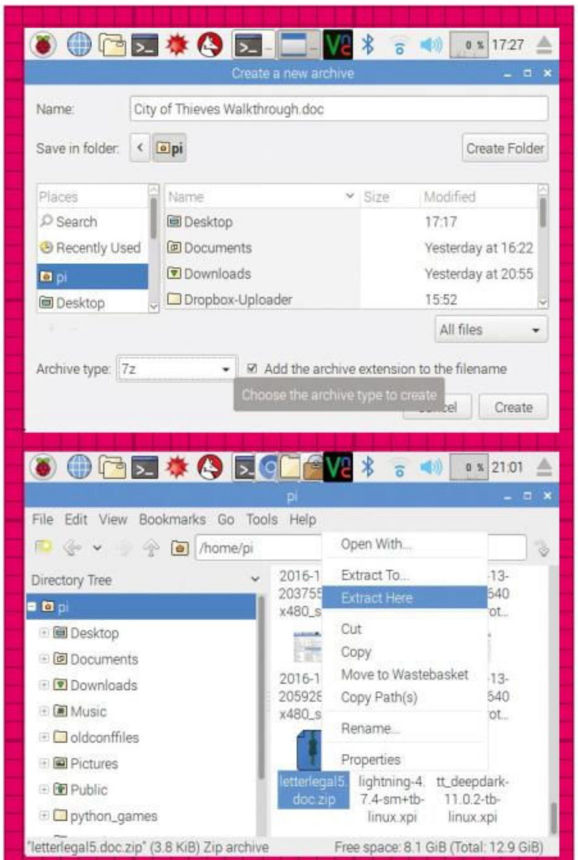
Create Zip files with a few mouse clicks

By default the Pi is capable of opening a range of compressed files such as Zip files. Simply right-click the file and choose Extract Here. As you use the Pi more, especially for email, you may well wish to compress files as well. The Pi supports various Linux formats out of the box such as .bz2 and .tar, but these are not particularly useful if you wish to send a file or folder to Windows or Mac users. So install support for Zip compression by clicking Preferences > Add/Remove Software and searching for "zip". The official name is "Archiver for Zip Files".

Once it's installed, you can right-click on a file or folder and choose Compress. The system will ask you where you want to save the Zip file. Choose a location, then click Archive Type at the bottom left and scroll to the bottom to select Zip. Now click Create at the bottom right. The Xarchiver program will display a dialogue stating "Adding to Archive" and the archive file will appear.

While adding support for Zip files, you may also wish to allow the Pi to create and extract compressed files using the 7Zip (.7z) format, which is more efficient and supports options such as splitting files and encryption. Search for "p7zip" in the Add/Remove Software window to do this. Once it's installed you will see .7z available under Archive Type when you're compressing a file.

»


Create a new archive

Name: City of Thieves Walkthrough.doc

Save in folder: pi

Archive type: 7z

☑ Add the archive extension to the filename

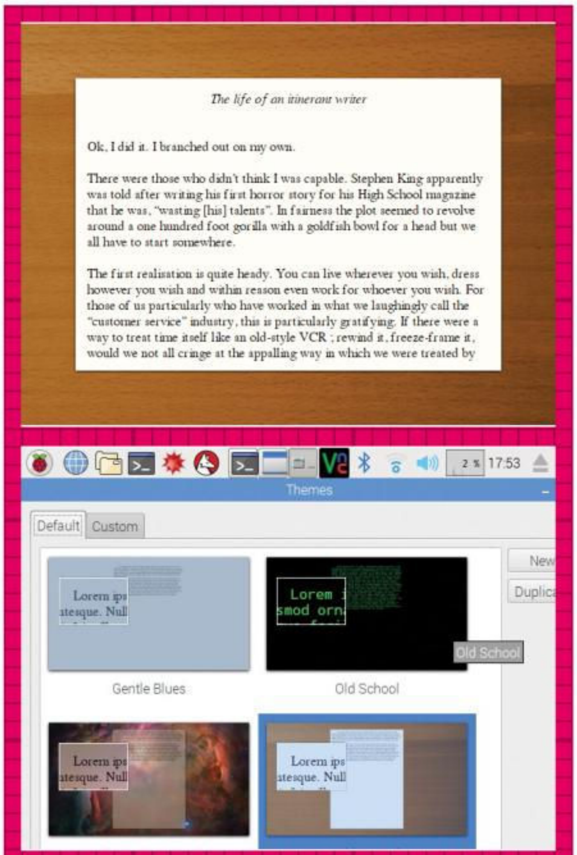
Choose the archive type to create

"letterlegal5.doc.zip" (3.8 KiB) Zip archive

Free space: 8.1 GiB (Total: 12.9 GiB)

17:27
Create Folder
Places
Search
Recently Used
pi
Desktop
Name
Size
Modified
Desktop 17:17
Documents Yesterday at 16:22
Downloads Yesterday at 20:55
Dropbox-Uploader 15:52
All files
Cancel
Create
21:01
pi
File Edit View Bookmarks Go Tools Help
/home/pi
Directory Tree
pi
Desktop
Documents
Downloads
Music
oldconffiles
Pictures
Public
python_games
Open With...
Extract To...
Extract Here
Cut
Copy
Move to Wastebasket
Copy Path(s)
Rename...
Properties
letterlegal5.doc.zip
lightning-4.7.4-sm+tb-linux.xpi
tt_deepdark-11.0.2-tb-linux.xpi

Install p7zip as well as Zip Archiver for compressing and extracting using the ultra-efficient 7Zip format.



The life of an itinerant writer

Ok, I did it. I branched out on my own.

There were those who didn't think I was capable. Stephen King apparently was told after writing his first horror story for his High School magazine that he was, "wasting [his] talents". In fairness the plot seemed to revolve around a one hundred foot gorilla with a goldfish bowl for a head but we all have to start somewhere.

The first realisation is quite heady. You can live wherever you wish, dress however you wish and within reason even work for whoever you wish. For those of us particularly who have worked in what we laughingly call the "customer service" industry, this is particularly gratifying. If there were a way to treat time itself like an old-style VCR ; rewind it, freeze-frame it, would we not all cringe at the appalling way in which we were treated by

Choose from several display options. Go to Settings > Focus Text to highlight individual lines or paragraphs.

13

FocusWriter

Pen a novel from the comfort of your Pi

Although the Pi does come with LibreOffice Writer pre-installed, those who write for the love of it may find the options rather overwhelming, although not nearly as much as the urge to quickly check on Facebook while working. FocusWriter is so named because it runs as a full screen app, specifically designed to help you ignore distractions and get on with writing.

Search for FocusWriter in the search bar in the Add/Remove Software window. After installing it you can launch the app from Applications > Office > FocusWriter. Go to Settings > Themes to customise the appearance of your writing area. The "Old School" theme strongly resembles the scrolling green lines of text from The Matrix. There is also a more traditional "Writing Desk" theme.

Although FocusWriter is suitable for any kind of writing, it's clear that it has been made with those who love to write in mind. The Tools menu has timers where you can set yourself goals as well as monitor your daily progress in terms of time spent writing or the number of words written. The Settings menu allows you enable Focus Text to highlight one or three lines, or indeed an entire paragraph.

The program can save documents in plain text (.txt), Rich Text (.rtf) and the open .odt format. And did we mention that FocusWriter makes typewriter sounds as you write?

14

USB Over IP

Access USB drives on your network

USB Over IP is a convenient way to access drives over a network. Once it's installed on the Pi and set up, any device connected to the network will be able to see and modify files as if the drive were plugged into that device.

To install it, search for "usbip" in the Add/Remove Software window. The software also has to be installed on any client devices that are going to access the USB drive. Versions for both Windows and Linux are available from **http://usbip.sourceforge.net**.

Once installed, it needs only a few simple commands to start. Open Terminal or connect via SSH and start the usbip server with the following commands:

```
sudo modprobe usbip-core
sudo modprobe usbip-host
sudo usbip -D
```

Next connect your USB drive, and use the following command to find its "busid":

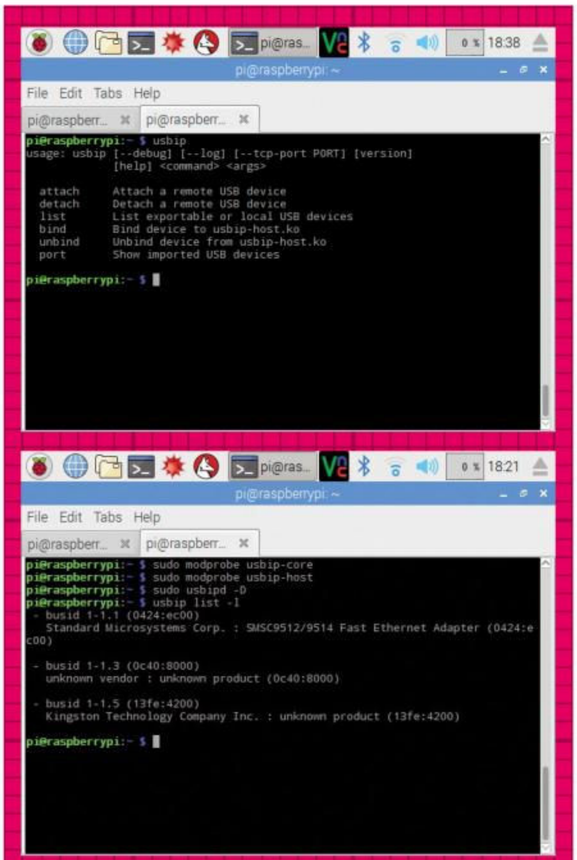
```
usbip list -l
```

A busid will be something like 1-1.5. Bind it to your Pi with:

```
sudo usbip --debug bind -b
(yourbusidhere)
```

Once USBIP is installed on your client device, you can run one command from the Windows or Linux command line to make the USB drive accessible over the network. See the SourceForge page for specific steps.

Be aware that the USB device will be available to anyone who is connected to your network and running the software. It also doesn't encrypt your data, so use it only on trusted networks.



```
pi@raspberrypi:~ $ usbip
usage: usbip [--debug] [--log] [--tcp-port PORT] [version]
             [help] <command> <args>

  attach     Attach a remote USB device
  detach     Detach a remote USB device
  list       List exportable or local USB devices
  bind       Bind device to usbip-host.ko
  unbind     Unbind device from usbip-host.ko
  port       Show imported USB devices

pi@raspberrypi:~ $ 
```

```
pi@raspberrypi:~ $ sudo modprobe usbip-core
pi@raspberrypi:~ $ sudo modprobe usbip-host
pi@raspberrypi:~ $ sudo usbipd -D
pi@raspberrypi:~ $ usbip list -l
 - busid 1-1.1 (0424:ec00)
   Standard Microsystems Corp. : SMSC9512/9514 Fast Ethernet Adapter (0424:e
c00)

 - busid 1-1.3 (0c40:8000)
   unknown vendor : unknown product (0c40:8000)

 - busid 1-1.5 (13fe:4200)
   Kingston Technology Company Inc. : unknown product (13fe:4200)

pi@raspberrypi:~ $ 
```

Your Bus ID may vary from those shown here. Use the **list** command to discover your own device's ID.

15

DOSBox

Play the DOS classics with one handy app

If you grew up in the '90s, you'll remember DOS classics such as Doom, Command and Conquer, SimCity and the Ultima series. Whether you're nostalgic or just curious, you'll find that few machines are capable of playing old DOS games natively. DOSBox solves this problem by emulating old Intel x86 PCs, right down to the DOS-like command prompt.

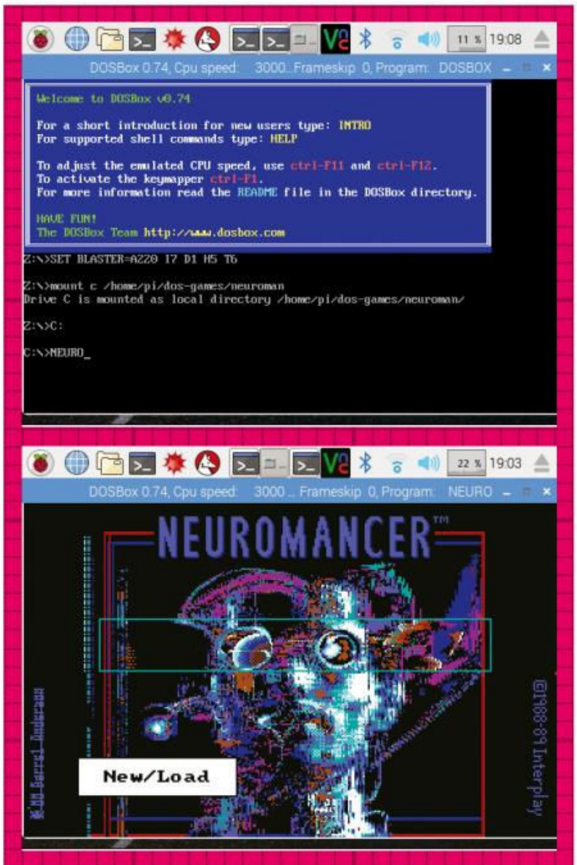
To install DOSBox, start by searching for "dosbox" in the search pane of the Add/Remove Software window. After installing it, create a folder in your home folder named **dos-games** to store your titles.

Where can you find these? One of the best known sites for DOS classics is **www.abandonia.com**. Abandonware is software no longer sold or supported by the copyright holder, meaning prosecution is unlikely, but do check that downloading games is legal in your jurisdiction.

Once you've downloaded the Zip file of a game, extract it and copy the folder to **/home/pi/dos-games**. Launch DOSBox from Applications > Games > DOSBox Emulator. Now you have to mount the game. For instance to mount Neuromancer in the folder neuroman as drive C, type

```
mount c /home/pi/dos-games/
neuroman
```

Next, to access the drive, type `C:`, then `dir` to list the files. To run the game, simply type the name of the .EXE file, for example NEURO. Press Ctrl+F1 inside any game to load the key mapper.


DOSBox 0.74, Cpu speed: 3000 Frameskip 0, Program: DOSBOX
DOSBox 0.74, Cpu speed: 3000 Frameskip 0, Program: NEURO
NEUROMANCER
New/Load
©1988-89 Interplay

```
Welcome to DOSBox v0.74

For a short introduction for new users type: INTRO
For supported shell commands type: HELP

To adjust the emulated CPU speed, use ctrl-F11 and ctrl-F12.
To activate the keymapper ctrl-F1.
For more information read the README file in the DOSBox directory.

HAVE FUN!
The DOSBox Team http://www.dosbox.com
Z:\>SET BLASTER=A220 I7 D1 H5 T6

Z:\>mount c /home/pi/dos-games/neuroman
Drive C is mounted as local directory /home/pi/dos-games/neuroman/

Z:\>C:

C:\>NEURO_
```

Neuromancer was all about the Matrix before the Matrix was even cool. Relive the glory on your Pi today.

16

Guake

Become a Terminal virtuoso

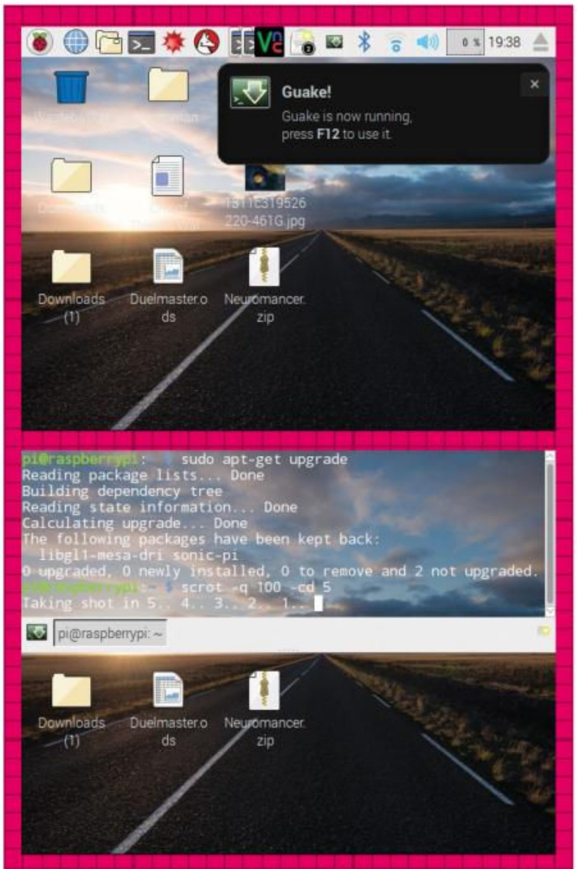
No matter how accustomed you are to graphical desktops and pointing-and-clicking, inevitably there comes a time with the Raspberry Pi where you will come to accept that using the Terminal is much easier. The Pi's built-in Terminal is adequate but it can obscure your view of the screen, and the black wall of text can be foreboding to newcomers. Guake tries to address these issues and others by being easier on the eye and much simpler to use.

Guake is technically as a "top-down" terminal. Install it via the Add/Remove Software window, then launch it from Applications > System Tools > Guake Terminal. A popup verifies that Guake is running. By default you can tap F12 to have the Guake terminal drop down. Enter your commands exactly as you would using the regular Terminal app.

It's much easier to pull up and dismiss, and you'll also note that the window is semi-transparent, so you can see any apps running in the background.

Right-click inside the Guake window to set its Preferences. These range from altering the look and feel of the window to changing the keyboard shortcuts to show and hide Guake.

Another huge advantage of Guake for budding programmers is that it can open tabs which are preconfigured to run Python programs. Usefully, Guake can also open a new tab in the current working directory. »



Guake is now running, press **F12** to use it.

```
pi@raspberrypi: ~ $ sudo apt-get upgrade
Reading package lists... Done
Building dependency tree
Reading state information... Done
Calculating upgrade... Done
The following packages have been kept back:
  libgl1-mesa-dri sonic-pi
0 upgraded, 0 newly installed, 0 to remove and 2 not upgraded.
pi@raspberrypi:~ $ scrot -q 100 -cd 5
Taking shot in 5.. 4.. 3.. 2.. 1.. 
```

If you've played the game Quake you'll recognise Guake's similarity to the game's own chat terminal.

17

Calibre

Convert, organise and share your ebooks

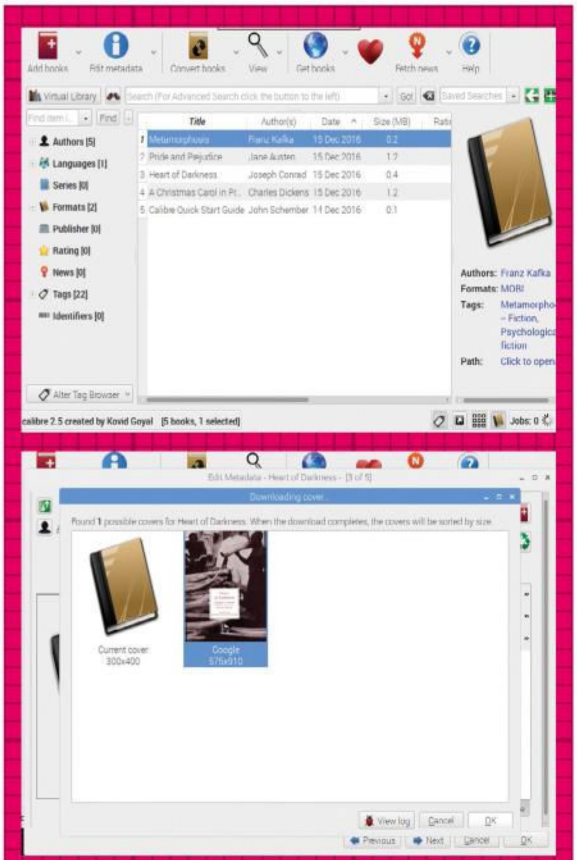
Calibre describes itself as a "complete library solution". It is perfect for anyone with a device capable of reading ebooks such as the Amazon Kindle.

Install Calibre via the Add/Remove Software window, then launch it from Applications > Office > Calibre.

On first run Calibre will ask you some questions about your e-book device and the format it supports. For instance the Kindle can read both the open .mobi format and Amazon's own .awz3. You will then be able to add books to your library. A huge number of formats are supported. Before converting into a format compatible with your device, you can right-click on one or more books to edit the metadata. This handy feature allows you to type in the correct author name and title. Calibre can also search online for the correct book cover and a plot summary.

Once a book has been converted, you can then either save it to a different folder on the Pi or, if your device is connected, effortlessly send it to the device's main memory using the buttons at the top of the Calibre window.

Another remarkably useful feature is Calibre's ability to run as an e-book server, which can be accessed by anyone via a web browser. Once your library has been set up, the Pi no longer needs a monitor, so you can run it "headless" if you wish.


Add books
Edit metadata
Convert books
View
Get books
Fetch news
Help
Virtual Library
Authors [5]
Languages [1]
Series [0]
Formats [2]
Publisher [0]
Rating [0]
News [0]
Tags [22]
Identifiers [0]
Alter Tag Browser
Authors: Franz Kafka
Formats: MOBI
Path: Click to open
Edit Metadata - Heart of Darkness - [3 of 5]
Downloading cover...
Current cover 300x400
Google 576x910
View log
Cancel
OK
Previous
Next

	Title	Author(s)	Date	Size (MB)
1	Metamorphosis	Franz Kafka	15 Dec 2016	0.2
2	Pride and Prejudice	Jane Austen	15 Dec 2016	1.7
3	Heart of Darkness	Joseph Conrad	15 Dec 2016	0.4
4	A Christmas Carol in Pr...	Charles Dickens	15 Dec 2016	1.2
5	Calibre Quick Start Guide	John Schember	14 Dec 2016	0.1

calibre 2.5 created by Kovid Goyal [5 books, 1 selected]

Found 1 possible covers for Heart of Darkness. When the download completes, the covers will be sorted by size.

▲ In the Calibre web interface, click on the cover of any book to download to your device.

18

Arduino IDE

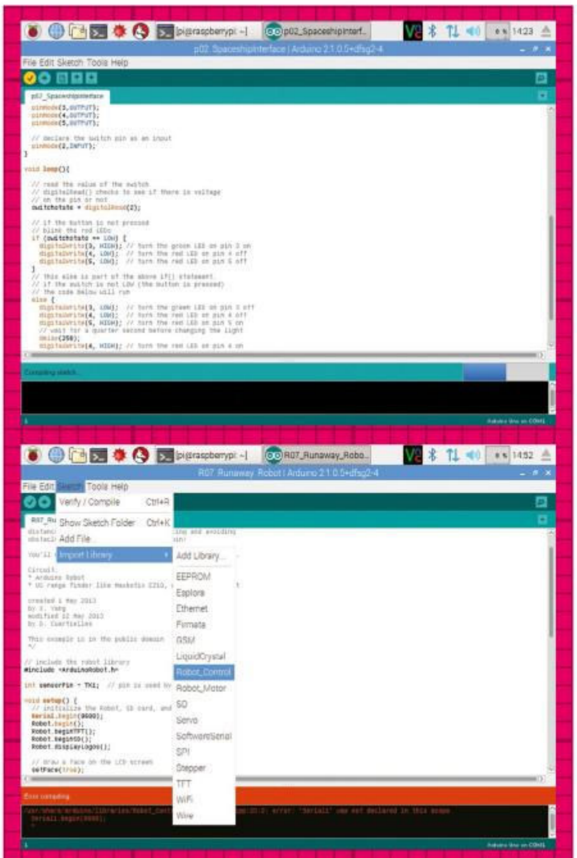
Write code and upload to an Arduino board

The Arduino is a "microcontroller motherboard". Microcontrollers are generally used to perform repeated actions – opening doors, checking temperatures and so on. Arduino projects are fantastic ways to get into both coding and electronics. While you can do this with the Pi itself, the Pi is a full-blown computer, much more than is needed for most such applications.

The Arduino IDE (Integrated Development Environment) makes it very easy to write code and upload it to any Arduino connected to the Pi. This is usually done via USB, though some boards can connect wirelessly. By default the software is preconfigured for the default Arduino Uno board, but you can go to Tools > Boards to select from the dozens available.

Install Arduino IDE using Add/Remove Software by searching for "arduino". To use it, go to Applications > Programming > Arduino IDE. To start, you can browse exciting sample projects by clicking File > Examples.

Like most programming platforms, Arduino can be extended further using Libraries. These provide extra functions when working with new hardware or processing data. For instance the LiquidCrystal library enables you to control connected Liquid Crystal Displays. See **www.arduino.cc/en/Reference/Libraries** for a full rundown of the official Arduino libraries.


pi@raspberrypi: ~
p02_SpaceshipInterf...
p02_SpaceshipInterface | Arduino 2:1.0.5+dfsg2-4
File Edit Sketch Tools Help
R07_Runaway_Robo...
R07_Runaway_Robot | Arduino 2:1.0.5+dfsg2-4
Verify / Compile Ctrl+R
Show Sketch Folder Ctrl+K
Add File...
Import Library
Add Library...
EEPROM
Esplora
Ethernet
Firmata
GSM
LiquidCrystal
Robot_Control
Robot_Motor
SD
Servo
SoftwareSerial
SPI
Stepper
TFT
WiFi
Wire
Arduino Uno on COM1

▲ To do more with a connected Arduino board, click Add Library to import libraries downloaded from the internet.

19

YouTube Downloader

Download video clips from streaming sites

Although technically it’s possible to use VLC Player (see page 14) to download and record video streams, the command line app youtube-dl makes it much easier to download videos from streaming websites. The name is something of a misnomer as it is actually compatible with several video sharing websites.

To install it, go to Applications > Add/Remove Software and search using the bar at top left for “youtube-dl”.

Once installed it can be launched at any time using Terminal on your Pi. The basic syntax for downloading video is:

```
youtube-dl -o [name of output
file] [YouTube URL]
```

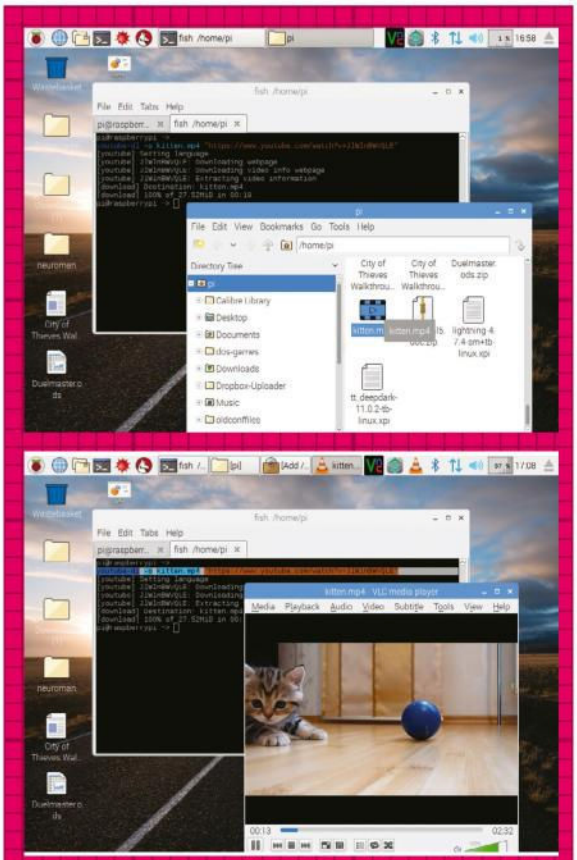
For instance, to download a video of Paris the fluffy kitten from the YouTube channel “funnycatsandnicefish” as an MP4 file, you would enter:

```
youtube-dl -o kitten.mp4
"https://www.youtube.com/
watch?v=JlWlnBWVQLE"
```

Be warned that the Pi may struggle to play high-definition videos. These will also take up precious space. You can tweak the quality of your downloads with the **-f** (format) flag. For instance, to ensure that the video of Paris the fluffy kitten is no more than 25MB in size, enter the above command like so:

```
youtube-dl -o kitten.mp4 -f
'best[filesize<25M]' "https://
www.youtube.com/
watch?v=JlWlnBWVQLE"
```

For a full rundown of all youtube-dl commands visit **https://github.com/rg3/youtube-dl/blob/master/README.md**



```
[youtube] Setting language
[youtube] JlWlnBWVQLE: Downloading webpage
[youtube] JlWlnBWVQLE: Downloading video info webpage
[youtube] JlWlnBWVQLE: Extracting video information
[download] Destination: kitten.mp4
[download] 100% of 27.52MiB in 00:19
```

fish /home/pi
File Edit Tabs Help
File Edit View Bookmarks Go Tools Help
/home/pi
Directory Tree
Calibre Library
Desktop
Documents
Downloads
Music
kitten.mp4 - VLC media player
Media Playback Audio Video Subtitle Tools View Help

Download video clips using youtube-dl, then use the indispensable VLC Player to play them.

20

Xpad

Smother your desktop with sticky notes

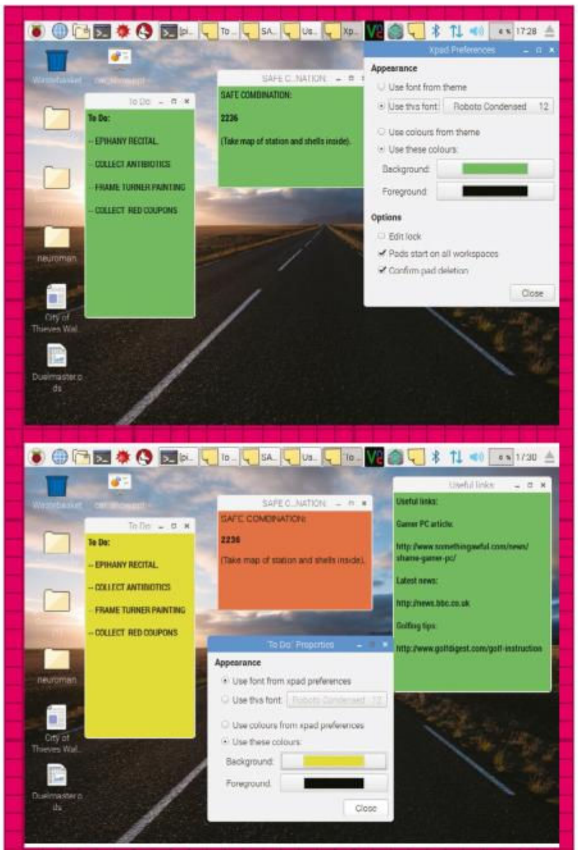
Xpad, which comes bundled with some versions of Linux such as Lubuntu, is a handy app for jotting down quick notes.

To install it, go to Applications > Preferences > Add/Remove Software and search for “xpad” using the bar at the top right. Once it’s installed, launch Xpad from Applications > Accessories.

Superficially Xpad windows resemble Post-it Notes (other brands of sticky label are also available), but they can hold unlimited amounts of text. Right-click anywhere inside a window and choose Edit > Preferences to change global settings. This will enable you to change the font, as well as the foreground colours. You can also choose to lock your Xpad windows for editing if they contain important information like vital internet links.

You can customise individual notes even further by right-clicking on them individually and selecting Properties. From here you can choose to use the global Xpad theme or colour code a note as you see fit. Click the + button at the bottom left of any pad to create a new note.

Right-click on the Xpad tray icon or any note to see the Note’s category. Each is named after the first line of text – for example “To Do”. Deleting notes is just as simple. Just click on the wastebasket icon on any pad. Simply closing a note will not delete it, meaning you can have more notes than the available space on your desktop. ■



To Do:

-- EPIPHANY RECITAL.

-- COLLECT ANTIBIOTICS

-- FRAME TURNER PAINTING

-- COLLECT RED COUPONS

SAFE COMBINATION:

2236

(Take map of station and shells inside).

Useful links:

Gamer PC article:

http://www.somethingawful.com/news/shame-gamer-pc/

Latest news:

http://news.bbc.co.uk

Golfing tips:

http://www.golfdigest.com/golf-instruction

Xpad Preferences

Appearance

○ Use font from theme

● Use this font: Roboto Condensed 12

○ Use colours from theme

● Use these colours:

Background: 🟩

Foreground: ■

Options

☐ Edit lock

✓ Pads start on all workspaces

✓ Confirm pad deletion

'To Do' Properties

Appearance

● Use font from xpad preferences

○ Use this font: Roboto Condensed 12

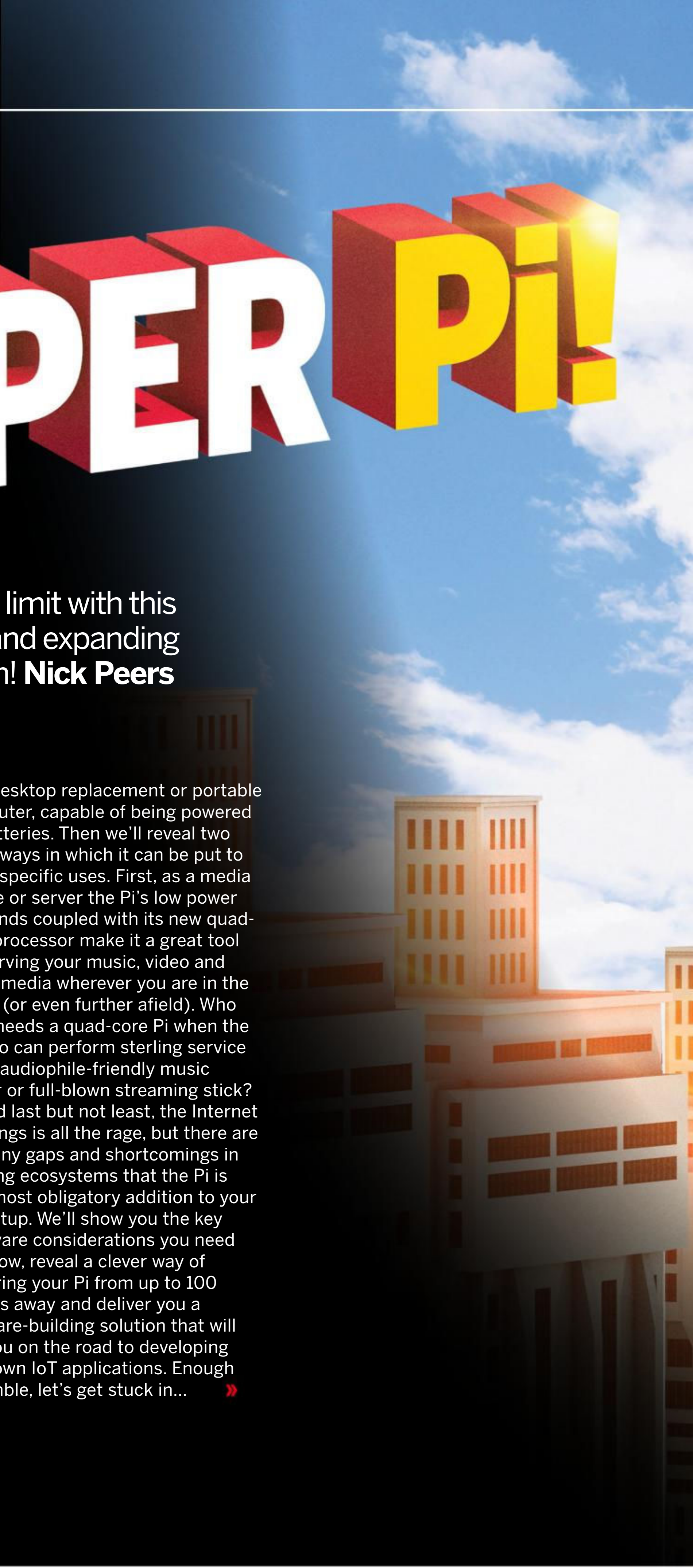
○ Use colours from xpad preferences

● Use these colours:

Background: 🟨

Foreground: ■

We don’t know if information wants to be free, but it’s useful to have it to hand in customisable sticky notes.



Build a SUPER Pi!

Push your Raspberry Pi to the limit with this essential guide to enhancing and expanding the best tiny PC known to man! **Nick Peers** is your gamma ray source

The Raspberry Pi has transformed the way we look at computers and computing. Thanks to its ridiculously low cost of ownership, you can build a fully functional computer for under a tenner and put it to all kinds of uses. Sometimes that potential can be so open as to seem overwhelming – this is a computer that has gone into space, for goodness sake – but that shouldn't stop you finding new and exciting ways to use it.

The Pi's versatility makes it the perfect tool for just about any task you could think for it to do. Looking for a desktop computer to replace your PC? The Raspberry Pi 3 has just about enough oomph – with the right support – to do that. How about using it as a portable device? A laptop maybe? Or at the very least as a portable touch-friendly unit? No problem.

In this feature, we explore how to build the ultimate Pi for use in a number of essential and common roles. We'll show you how to set it up as a desktop replacement or portable computer, capable of being powered by batteries. Then we'll reveal two great ways in which it can be put to more specific uses. First, as a media centre or server the Pi's low power demands coupled with its new quad-core processor make it a great tool for serving your music, video and other media wherever you are in the home (or even further afield). Who even needs a quad-core Pi when the Pi Zero can perform sterling service as an audiophile-friendly music player or full-blown streaming stick?

And last but not least, the Internet of Things is all the rage, but there are so many gaps and shortcomings in existing ecosystems that the Pi is an almost obligatory addition to your IoT setup. We'll show you the key hardware considerations you need to follow, reveal a clever way of powering your Pi from up to 100 metres away and deliver you a software-building solution that will put you on the road to developing your own IoT applications. Enough preamble, let's get stuck in... »

This aluminium case from FLIRC has a built-in heatsink, which works wonderfully.

The ultimate desktop Pi

Squeeze all the juice out of the Raspberry Pi for a desktop PC

The biggest advantage of using the Pi at the heart of a desktop-based system is its low cost. But if you want to eke every last drop of performance from it, what do you need to do? The first thing is – of course – to choose the latest Raspberry Pi 3 model as the core of your desktop. It's the fastest, most powerful Pi yet, and you can squeeze additional power from it through overclocking – setting the processor to a higher speed (we'll explain more over the page).

Overclocking raises the next issue: how can you keep your Pi from overheating? The solution is to add a heatsink. You can purchase these for as little as £1 each from the likes of Pimoroni, but choose carefully: the 7.5mm sink will dissipate that bit more heat, but you won't be able to fit a HAT on top of it. If this is a deal-breaker, choose the 6mm model. If you want to push things even further, then check out ModMyPi's heatsink pack. It costs £4, and pairs a 14mm-high sink for the CPU with a 10mm-high sink for the Ethernet controller. Again, these should fit into most standard cases, but won't play nicely with any HATs.

An alternative is to look at the FLIRC case (£13, **www.pi-supply.com**). Its aluminium shell includes a built-in heatsink that basically protrudes from the shell to sit on top of the CPU and draw heat away from it. The result is that your overclocked Pi 3 won't max out at more than 65 degrees Celsius, but again there's no room for any HATs.

One final recommendation is to shell out for the latest version of the official Raspberry Pi 3 power supply. It should supply 5.1V and 2.5A of power, as compared to the 5V/2A offered by older and unofficial models. Doing so will give you all the juice you need to push your Pi that bit further.

A TRUE DESKTOP PI

If you want your Pi to form the heart of an all-in-one device, then visit **www.pi-top.com** to learn more about the Pi-topCEED, a desktop-friendly chassis with built-in 14-inch HD screen (1,366x768 resolution) and space behind for your Raspberry Pi (you can purchase it with or without a Pi 3).

Fitting is incredibly easy – remove the front panel from beneath the display to reveal a generous space for your Pi and any add-on boards you want to include. Hook up the Pi to the HDMI cable, place it on the mag rail and slide it to the end, where it clips securely into place. HATs and other add-ons are easy to install too – you'll need to purchase a pi-topPROTO accessory, which plugs into the mag rail and slides into place on it. There's also an optional speaker component for adding sound.

You can pick up a PiTop CEED Pro in green or grey from £119 including VAT from RS Components (**https://uk.rs-online.com**). It's undoubtedly an attractive way to house your desktop Pi, although the screen is a little small.

An attractive way to house a desktop Pi, although the screen is a little small.

The Pi 3 has one other major advantage over earlier models: it supports USB booting. This means that with a bit of work (see *Run Ubuntu Mate from USB*, over the page) you can pair it with a USB flash or SSD hard drive to deliver a more responsive experience. Some users have reported success powering SSD drives from the Pi 3's USB port using a Y-splitter cable, but we found the drive was unstable and threw up errors when creating the root filesystem. If you fall into this category, you'll need to power the drive separately using its own dedicated supply.

If you're in the market for a suitable USB drive, the obvious choice is the PiDrive range from Western Digital (**http://wdlabs.wd.com**) if you're after a larger hard drive, but SSD and flash drives will produce better performance. Despite the USB 2.0 bus, it appears USB 3.0 flash drives produce better results – the SanDisk Cruzer Ultra and Extreme 32GB and 64GB drives have been reported as both compatible and recommended thanks to their superior performance.

If you'd like your Pi to turn heads, then check out the *True Desktop Pi* box opposite for details of an all-in-one chassis that includes a 720p HD display. Speaking of displays, the Pi 3 can output 4K on a suitable monitor, but you'll need to tweak the **config.txt** file to add a suitable mode at 15Hz:

```
hdmi_cvt 3840 2160 15
```

You may also want to consider another distro in place of the standard Raspbian – Ubuntu Mate (**https://ubuntu-mate.org/raspberry-pi**) is a slick alternative, but is a bit more demanding. We cover both in the next section.

The clever Pi-topCEED turns your Pi into an all-in-one desktop computer – see opposite page.

Boost performance

Want to use your Pi as a desktop replacement? Here's what you need to do

You've purchased your kit and you're ready to build your ultimate desktop system. First things first: which distro will you choose? The obvious one is the default Raspberry Pi operating system Raspbian, but if you're put off by its rather basic interface, then Ubuntu Mate should appeal. It provides a slicker, more desktop-like environment, but isn't as well supported and there's a performance price to pay for those bells and whistles – a slight trade-off between desktop-like appearance and desktop-like performance – although you're less likely to feel the pain with the Pi 3. If Ubuntu Mate sounds like it could be your kind of thing, check out the *Run Ubuntu from USB* box over the page.

Boot into Raspbian from USB

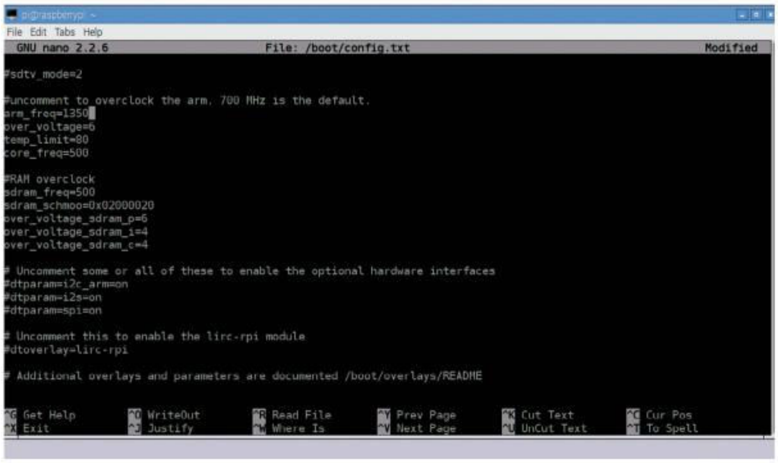
Instructions for preparing a Pi 3 for booting from USB 3.0 can be found at **www.raspberrypi.org/documentation** under Hardware > Raspberry Pi > BootModes > Mass Storage Device. To put it simply, you'll first update your Pi's firmware with the experimental boot files, then enable USB boot mode. Reboot to test, then connect your USB drive and use **parted** to format it correctly with a 100MB FAT32 partition and Linux ext4 partition. You then create the boot and root filesystems, mount the new filesystems and copy your Raspbian system to it. Regenerate SSH host keys, edit the boot files to look for the USB drive, unmount your filesystems and then power off your Pi. Remove the SD card, reconnect the power and if all is well you'll be booting from your USB drive.

Sadly, the process isn't guaranteed to work with all drives – we were unable to get our Kingston SSDNow 64GB drive mounted in a StarTech chassis to boot directly as a result. We thought the problem might be down to the need for a longer delay at startup to allow the drive to be detected, so followed the workaround at **www.raspberrypi.org/documentation/hardware/raspberrypi/bootmodes** which involves formatting a SD card (any size will do) in FAT32 and placing an updated **bootcode.bin** file on it along with an empty file you name **timeout**. With this in place, we were able to run Raspbian from the USB drive.

You may or may not notice an improvement in boot times, but you should quickly find that your Pi is a little more responsive, which you can confirm by benchmarking the drive using `dd` and `hdparm`:

```
$ dd if=/dev/zero of=~/test.tmp bs=500K
count=1024
$ sudo hdparm -t /dev/sda
```

By using `dd`, we get the drive's sequential write speed by writing a 500KB file 1,024 times, while `hdparm` provides the drive's sequential read speed. We performed each test five times to get an average score – the SSD drive managed 30MBps »



```
  GNU nano 2.2.6                  File: /boot/config.txt                         Modified

#sdtv_mode=2

#uncomment to overclock the arm. 700 MHz is the default.
arm_freq=1350
over_voltage=6
temp_limit=80
core_freq=500

#RAM overclock
sdram_freq=500
sdram_schmoo=0x02000020
over_voltage_sdram_p=6
over_voltage_sdram_i=4
over_voltage_sdram_c=4

# Uncomment some or all of these to enable the optional hardware interfaces
#dtparam=i2c_arm=on
#dtparam=i2s=on
#dtparam=spi=on

# Uncomment this to enable the lirc-rpi module
#dtoverlay=lirc-rpi

# Additional overlays and parameters are documented /boot/overlays/README


^G Get Help     ^O WriteOut     ^R Read File    ^Y Prev Page    ^K Cut Text     ^C Cur Pos
^X Exit         ^J Justify      ^W Where Is     ^V Next Page    ^U UnCut Text   ^T To Spell
```

To overclock your Pi 3 you'll need to edit the /boot/config.txt file – it can't be done from the raspi-config tool.

RUN UBUNTU MATE FROM USB

If you want to run Ubuntu Mate from your USB drive, then you will need to follow a different procedure. The step-by-step guide (below) reveals the prerequisites—you basically copy the Ubuntu Mate installer to your USB drive and copy the boot partition to a freshly formatted micro SD card for the initial installation.

All being well, you should be able to boot from your micro SD card, which then switches to the USB drive to complete the installation. Once you're logged in properly, open a Terminal window and type the following command:

```
$ sudo fdisk /dev/sda
```

Type `p` and press Enter to make a note of where block **sda2** starts (typically it's 133120). Now type `d` and hit Enter, then choose 2 to delete the second partition. Press `n` and hit Enter to create a new partition and type `p` for Primary partition followed by `2`. Now enter the figure you noted earlier (133120 in our case) and hit Enter, then hit Enter again to fill the remaining space. Finally, press **w** and hit Enter to make the changes. You'll see the warning, so reboot by typing

```
sudo reboot
```

Once rebooted, open another Terminal window and complete the resizing process with:

```
$ sudo resize2fs /dev/sda2
```

Once resized, verify the change via `df -h` and then use `sudo blkid` to identify the drive's UUID. Next, `sudo nano /etc/fstab` to edit the line from **/dev/mmcblk0p2 to UUID=e440adac-fcf9-4b68-9f94-6bfd030f60b3** (or whatever your drive's UUID is). Save the file, reboot and you are finished. Note: You will still need to keep the micro SD inserted for Ubuntu Mate to boot, but it should be a bit more responsive, and you shouldn't have any problems booting even when other USB drives have been attached. We'd recommend that you visit **http://bit.ly/Pi2UbuntuMateSetup** as well for some other tweaks and tips to try out for optimising Ubuntu Mate.

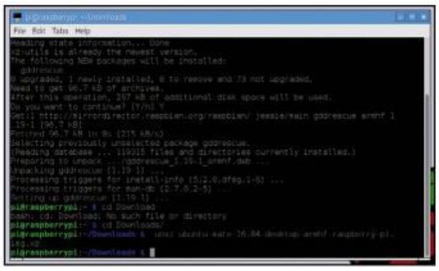
(write) and 33MBps (read), compared to 8MBps and 20MBps respectively for the SanDisk Ultra SD card. It's worth noting that these aren't true real-life tests – random read/write times will see a smaller gap between the two – but a fast USB drive will still be a step up, nonetheless.

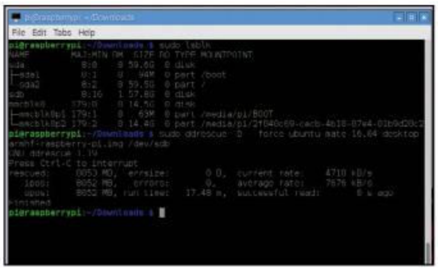
Overclock your Pi

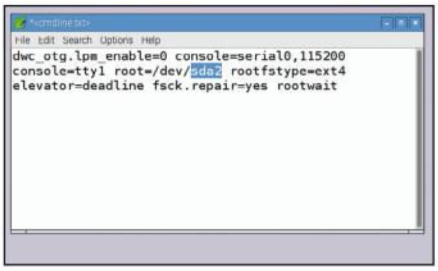
If you're looking to eke that bit more performance from your Pi, then obviously overclocking – that is, manually cranking the processor up to a higher speed than its default setting – is the way to go. Sadly not all Pi 3s are equal, and you can't overclock using the official Raspbian configuration tool (or raspi-config for that matter), but don't let that stop you, because you can easily overclock your Pi by editing the **/boot/config.txt** file.

There's actually three things you can overclock: the processor, obviously, but also your Pi's RAM and GPU too. A list of suggested settings for all these can be found at **https://github.com/retropie/retropie-setup/wiki/Overclocking**. We'd suggest sticking with the more modest 1.35GHz overclock and skip the GPU overclocking

INSTALL UBUNTU MATE TO YOUR USB DRIVE







```
dwc_otg.lpm_enable=0 console=serial0,115200
console=tty1 root=/dev/sda2 rootfstype=ext4
elevator=deadline fsck.repair=yes rootwait
```

1 Download and Install

First you'll need to go to the website **https://ubuntu-mate.org**, click the Download button and download the Raspberry Pi build to your internal storage. This will be in XZ compressed format, so open a Terminal window and type the following command:

```
$ sudo apt-get install gddrescue xz-uti
&& cd ~/Downloads
$ unxz ubuntu-mate-16.04-desktop-
armhf-raspberry-pi.img.xz
```

2 Write to USB drive

Insert your USB drive into your Pi and umount it if necessary. Now type the following to identify it:

```
sudo lsblk
```

For example, you'll see drives designated as **/dev/sdb** or **/dev/sdc**). Now type the following command, replacing the **/dev/sdc** designation in the example with the correct device name:

```
$ sudo ddrescue -D --force ubuntu-
mate-15.10.3-desktop-armhf-raspberry-
pi-2.img /dev/sdc
```

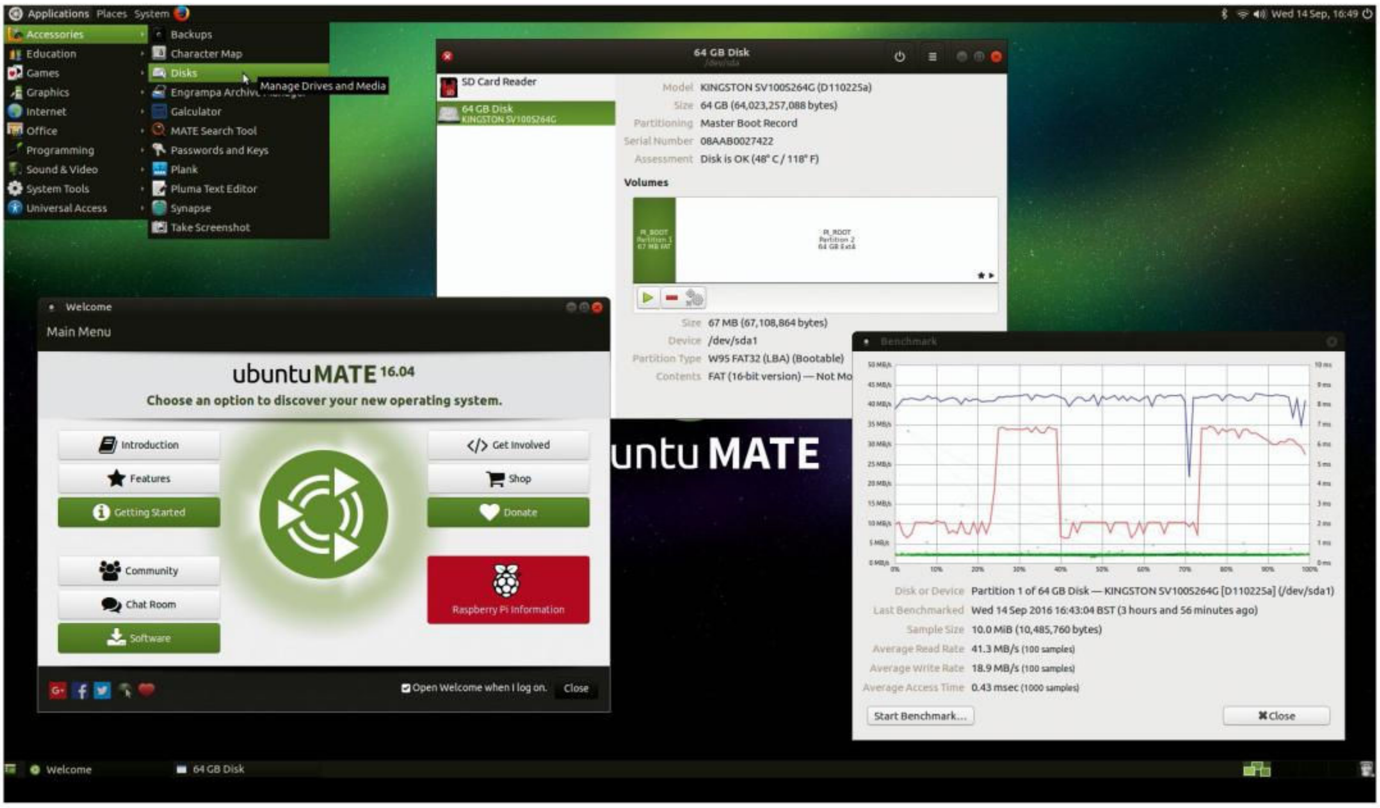
Now wait while the image is written to the drive.

3 Copy the root partition

Insert a spare micro SD card using a USB adaptor and format it using FAT32. Next, copy the contents of your USB drive's boot partition to the micro SD card. Now open the **cmdline.txt** file on the micro SD card and change the **/dev/mmcblkp2** reference to

```
/dev/sda2
```

Save the file and exit. Plug the USB drive into your Raspberry Pi, insert the microSD card and connect the power to start installation.


Applications Places System
Wed 14 Sep, 16:49
Accessories
Education
Games
Graphics
Internet
Office
Programming
Sound & Video
System Tools
Universal Access
Backups
Character Map
Disks
Engrampa Archive Manager
Manage Drives and Media
Calculator
MATE Search Tool
Passwords and Keys
Plank
Pluma Text Editor
Synapse
Take Screenshot
Welcome
Main Menu
ubuntu MATE 16.04
Choose an option to discover your new operating system.
Introduction
Features
Getting Started
Get Involved
Shop
Donate
Community
Chat Room
Software
Raspberry Pi Information
☑ Open Welcome when I log on.
Close
ubuntu MATE
SD Card Reader
64 GB Disk
KINGSTON SV100S264G
Volumes
PI_BOOT
Partition 1
67 MB FAT
PI_ROOT
Partition 2
64 GB Ext4
Benchmark
50 MB/s
45 MB/s
40 MB/s
35 MB/s
30 MB/s
25 MB/s
20 MB/s
15 MB/s
10 MB/s
5 MB/s
0 MB/s
10 ms
9 ms
8 ms
7 ms
6 ms
5 ms
4 ms
3 ms
2 ms
1 ms
0 ms
0%
10%
20%
30%
40%
50%
60%
70%
80%
90%
100%
Start Benchmark...
Close
Welcome
64 GB Disk

64 GB Disk
/dev/sda

Model KINGSTON SV100S264G (D110225a)
Size 64 GB (64,023,257,088 bytes)
Partitioning Master Boot Record
Serial Number 08AAB0027422
Assessment Disk is OK (48° C / 118° F)

Size 67 MB (67,108,864 bytes)
Device /dev/sda1
Partition Type W95 FAT32 (LBA) (Bootable)
Contents FAT (16-bit version) — Not Mo

Disk or Device Partition 1 of 64 GB Disk — KINGSTON SV100S264G [D110225a] (/dev/sda1)
Last Benchmarked Wed 14 Sep 2016 16:43:04 BST (3 hours and 56 minutes ago)
Sample Size 10.0 MiB (10,485,760 bytes)
Average Read Rate 41.3 MB/s (100 samples)
Average Write Rate 18.9 MB/s (100 samples)
Average Access Time 0.43 msec (1000 samples)

Ubuntu Mate can be run from a USB drive too, but you'll still need a microSD card for booting.

section (the **gpu-mem** setting refers to the amount of memory you allocate the graphics card – 128MB or even 64MB should be adequate for basic desktop use) unless you're looking to boost your gaming performance.

There are two things to consider. First, how do you verify that the overclock settings are in place? Use the following command to determine the minimum and maximum clock speeds:

```
$ lscpu | grep "MHz"
```

Second, how can you tell if your overclock is stable? Keep an eye on the CPU temperature by displaying it in the menu bar: right-click the menu bar and choose Add/Remove Panel Items. Click Add and select Temperature Monitor. You can then keep an eye on the temperature while you make use of your Pi. Anything under 80 degrees is fine, but with luck your cooling system will ensure that even when overclocked your CPU temperature remains under 60 degrees.

For a more in-depth look at overclocking, including some other tweaks you can apply (such as forcing your processor to always run in performance mode), check out Jackenhack's blog post at **www.jackenhack.com/raspberry-pi-3-overclocking**. There's a mention of running a custom Linpack script in the comments to stress-test your Pi, but we found that it locked up our Pi even without overclocking, and there's a long discussion about what this means at **http://bit.ly/LinPackStressTest**.

Desktop performance tips

Even with its quad-core processor, the Pi 3 isn't going to provide you with the same level of performance as even a modestly specified modern PC. One of the remaining bottlenecks is the 1GB of RAM – this is perfectly fine for one or two tasks, but you'll need to keep your aims realistic and avoid opening too many applications or browser tabs at once. Train yourself to keep things lean and mean, and you'll be able to enjoy perfectly adequate performance for basic desktop tasks such as web browsing (including using cloud applications) and office work.

One final tip: both Epiphany (Raspbian) and Firefox (Ubuntu) browsers struggle to render certain websites, particularly those loaded up with JavaScript. The simplest solution is to switch browser to Chromium, particularly if you're planning to make use of Google services like Drive or Docs. Ubuntu Mate users can install Chromium through the Software Center, but it's not available through normal means to Raspbian Jessie users. No matter – just open a Terminal and enter the following to download and install it:

```
$ wget -qO - http://bintray.com/user/download
SubjectPublicKey?username=bintray | sudo apt-key
add –
$ echo "deb http://dl.bintray.com/kusti8/chromium-
rpi jessie main" | sudo tee -a /etc/apt/sources.list
$ sudo apt-get update
$ sudo apt-get install chromium-browser rpi-
youtube -y
```

YOU CAN OVERCLOCK THE PROCESSOR, OBVIOUSLY, BUT ALSO YOUR PI'S RAM AND GPU

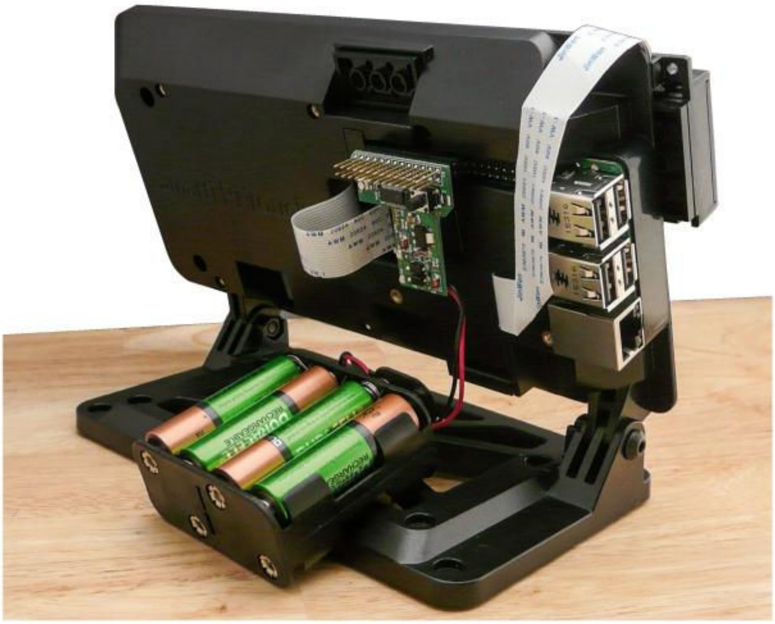
»

The ultimate portable Pi

Discover how to take your Pi on the road with a portable touchscreen

The Pi's small size and modest power demands make it the perfect candidate for use as a portable device, and thanks to the range of cases, displays and battery power options out there, you can configure it just about any which way you choose. We've decided to build our ultimate portable Pi around the official 7-inch touchscreen, but see the box below for alternatives you might want to consider.

The official touchscreen costs around £55 from any reputable Pi outlet. You'll also need to buy a suitable stand and case to attach your Pi to. There are plenty of options; we've plumped for the clever SmartiPi Touch case (£20, **www.modmypi.com**). The touchscreen fits snugly inside, secured by four screws, and there's a compartment on the back for your Pi too, with all key ports left accessible (the micro SD card is the only exception, but thanks to the design of the clickable cover that holds it in place, it's easy enough to take out, flip over and swap out the card).



Free your portable Pi from the mains supply by installing the MoPi add-on.

You can power the display one of two ways, the most convenient of which is via the 2-to-1 micro USB cable, which delivers power to both the Pi and the display from a single power cable. With this in mind, we recommend the newer 5.1V official power supply, particularly if you're planning to use HATs, which sit happily on top of the Pi's cover. (You can purchase an optional ribbon connector to extend them to the front of your Pi too.)

Smart screen

The only drawback with the SmartPi is that the Pi's own HDMI port is hidden away out of reach, which makes it impossible to connect up a second display. (If this is a deal breaker, examine an alternative such as the £15 black touchscreen case from The Pi Hut – visit **https://thepihut.com**.) All four USB ports and the Ethernet port are accessible, as is the 3.5mm audio jack for connecting to a speaker or headset. The case also comes with a LEGO-compatible camera case that clips neatly onto the front for video conferencing (and there's a handy slit in the compartment cover for slipping the camera cable through).

The case comes with an adjustable stand, which allows you to use it on any surface. More permanent fixtures are provided – there are 75mm VESA mount holes, and two small metal screw mounts are included – but for portability you'll be wanting to use the stand.

Construction is simple thanks to an online video; give yourself five to ten minutes and by the end your Pi will be snugly fitted along with the touchscreen, your camera mounted to the front and your HAT sitting happily on top of the compartment cover. If you're worried about overheating, the cover includes a knock-out for a 6mm CPU heatsink.

We also recommend downloading the latest version of Raspbian Jessie from **www.raspberrypi.org** (click Downloads > Raspbian), and once it's installed bringing it fully up to date with:

```
$ sudo apt-get update
```

ALTERNATIVE FORM-FACTORS

You don't need to build a portable Pi around the official 7-inch touchscreen. If you've got £225 to spare, you can purchase the Pi-Top, a full laptop chassis with keyboard, 13.3-inch HD screen (1,366x768) and smart battery offering up to 10 hours of life – see **www.pi-top.com/product/pi-top**.

Or you could go to the opposite extreme and experiment with a 3.5-inch resistive touchscreen. For a guide to building your own Nintendo DS sized computer with 3.5-inch display and keyboard, follow Chris Robinson's guide at **https://youtu.be/NMkhRK2IV0E** – it'll cost you under £100, but you'll need to be a bit of a whizz with a hacksaw blade and glue gun as well as a soldering iron.

A simpler – and cheaper – way to add a 3.5-inch display to your Pi is through SB Components' LCD Control Case (from £26, **shop.sb-components.co.uk**), which comprises a 320x240 resistive touchscreen with case, Wi-Fi dongle, tweaked Raspbian install (with LCD drivers) on an 8GB microSD card with a 3m long power supply. The main issue here is with the screen size – while it's technically possible to run the Raspbian desktop environment, you'll need to optimise it for the smaller screen. There's space for around six desktop icons, for example, and you may find it's more suited to Terminal use as well as mobile-friendly interfaces.

ADD RIGHT-CLICK TO THE TOUCHSCREEN

On the whole, the touchscreen works well as a mouse substitute, but it lacks right-click support. The solution given on the Raspberry Pi website works only in Wheezy, but a better solution exists in the form of twofing, which allows you to generate a right-click by tapping the screen using two fingers.

First, go to **http://plippo.de/p/twofing** for the basic instructions on downloading and compiling the package. Before rebooting as instructed, however, you need to perform some additional steps. First, create a **.rules** file:

```
$ sudo nano /etc/udev/rules.d/70-touchscreen-egalax.rules
```

Add the following line, then save and exit:

```
KERNEL=="event*",ATTRS{name}=="FT5406 memory based driver",SYMLINK+="twofingtouch",RUN+="/bin/chmod a+r /dev/twofingtouch"
```

```
$ sudo cp twofing /usr/bin/
```

Reboot, then test it with the following:

```
$ twofing --debug
```

If successful, this should launch the calibration screen. Simply press Ctrl+c to exit, then type `twofing` and hit Enter to launch it. Close the Terminal window and experiment with tapping the screen with two fingers and you should see the right-click menu pop up. Finally, to have twofing run automatically in future, add this line to **~/.config/lxsession/LXDE-pi/autostart**:

```
@/usr/bin/twofing
```

```
$ sudo apt-get upgrade
```

If you wish, you could also bring the firmware up to date using the following:

```
sudo rpi-update
```

This has no bearing on the performance of the touchscreen, although some users have reported that the Pi's CPU temperature drops by a couple of degrees through reduced load.

You can plug in a wired keyboard and mouse, but you might want to consider going down the wireless route; most wireless keyboards, including Ebuyer's own Xenta-brand range, should work fine. To save space, pick one with a built-in touchpad. You can, of course, interact with the touchscreen itself, but it's handy to have an alternative.

If you find the backslash key is missing from your wireless keyboard, don't panic – you can get the backslash character by holding the Alt Gr key and hitting **-**, and then the | (pipe) symbol is obtained by holding Alt Gr as you hit the **`** key in the top left-hand corner of the keyboard.

If you'd rather avoid using any form of wireless keyboard, you'll need to add an onscreen keyboard to Raspbian with:

```
$ sudo apt-get install Florence
```

Once installed, Florence is run from the Universal Access menu. You'll see the onscreen keyboard appear, plus a small floating keyboard tile, which you tap to bring the virtual keyboard to the screen and then dismiss it again. Sadly, it doesn't work well with the Terminal, so keep a USB keyboard to hand.

Elsewhere, the touchscreen works well as a mouse substitute, but there's no right-click support and the recommended solution on the Raspberry Pi website works only with Wheezy and not Jessie. However, there's a workaround that simulates a right-click using two fingers – see *Add Right-Click to the Touchscreen* above.

Portable power

As things stand, you have a compact, all-in-one solution for your Pi that's hamstrung by one critical problem: you need to be connected to mains power in order to use it. That's a problem that's easily fixed, and how you do this depends on your needs. The key consideration is that you supply a steady 5V of input, so you can't simply connect up four AA batteries and hope for the best – AA rechargeables would supply 4.8V in total, which would drop as the batteries discharged, while AA alkaline would supply 6V, which is outside the Pi's tolerance range.

The simplest solution is to power your Pi using any kind of portable battery that's used to recharge smartphones. Make sure they have a 5V output setting, and then all you need to do is plug one into your Pi to power it up.

But what size battery should you go for? It all depends on your needs. First, determine how much power (measured in milliamps per hour or mAh) you'll need to supply – not just to the Pi (up to 1,200mAh), but also to your touchscreen (around 500mAh) and any other devices (the camera requires 250mAh, for example). If you want to calculate the power draw of your USB devices, use the following command:

```
$ usb-devices | grep 'Product=\|MxPwr'
```

Tot up the figure and the result – 2,000mAh or upwards – is how much power you'll need for each hour of use. In theory this means a 10,000mAh battery will supply five hours of use, but in practice it's usually less because of the way they're manufactured, plus the fact that batteries' capacity drops over their lifetime.

An alternative way to supply portable power to your Pi is with the MoPi add-on (£25, **https://shop.pimoroni.com**). This plugs into your Pi's GPIO pins and enables you to connect a wide variety of batteries, from solar and car batteries to standard (and rechargeable) AA batteries, using standard screw terminals. There's also a handy on/off button, and the MoPi will cleanly shut down your Pi when power drops below a certain level. You can even use it in conjunction with mains power, which turns it into a handy UPS backup.

To keep things as portable as possible, we paired a MoPi with eight rechargeable AA batteries using a cheap AA battery holder. The MoPi supports input voltages from 6.2V up to 20V, so you'd need a minimum of six rechargeable 1.2V AA batteries, but we've gone for eight because you get longer battery life and it's one of the default supply profiles – all you need to do is remove the small jumper on the underside of the MoPi. Just follow the step-by-step guide (**https://pi.gate.ac.uk/pages/mopi.html**) and you'll soon have your portable Pi running independently of mains power. »

The ultimate media Pi

Build a Pi-based media centre, audio player or media server

There's one thing the Pi is brilliant at, and that's media. Whether you're looking for an audiophile-friendly music player, a simple set-top box for playing media through your TV, or a full-blown media server for accessing your digital media from anywhere, there's a Pi capable of delivering what you want. In this section, we're going to focus our efforts on creating a Pi media server, which will allow you to share all your video, music and photos with your other devices, both at home and further afield.

For the purposes of this project, we'll focus on the Raspberry Pi 3, largely because it delivers enough oomph to run a media server comfortably. But there are other benefits too – the Wi-Fi, in case you don't want to site it next to your router, and support for USB booting should you decide to dump the micro SD card in favour of an all-purpose drive for running your server as well as storing your media on.

The big choice you need to make is which media server to run. Kodi's built-in media server will work, of course, but the experience isn't as rich on non-Kodi devices in terms of delivering artwork and metadata to help you identify what you're watching or listening to. We've successfully run Plex Media Server on both a Pi 2 and a Pi 3 for a number of years now. If this appeals, check out **www.htpcguides.com** – go to Pi > Rasp Pi 2 and 3 > NAS and Media Server > Install Plex Media Server… for a suitable image and guide.

However, in the interests of flying the FOSS flag, we've plumped for Emby Media Server. It looks as good as Plex, although it isn't as well tested on the Pi. You'll find the basic steps required to get the server up and running in the guide at **https://akosresch.wordpress.com/2016/05/11/installing-emby-server-on-raspberry-pi-3**. You'll need to install imagemagick and the Mono library, and then it's a case of downloading and installing the Debian version of Emby Server.

Before launching Emby Server for the first time, we recommend doing the following, which will simplify issues accessing your media folders by making Emby run under your Pi user account:

```
$ sudo nano /etc/emby-server.conf
```

Add the following line to the end of the file before saving and closing:

```
EMBY_USER=pi
```

We'd also suggest you apply the FFmpeg fix. First type the following command:

```
sudo apt-get install git
```

Once that lengthy process is complete, type the following to start up Emby:

```
$ sudo service emby-server start
```

The server should now be up and running. Use the `ifconfig` command to determine your Pi's IP address, then switch to another computer and type the following into your web browser (replacing **192.168.x.y** with your Pi's IP address):

```
192.168.x.y:8096
```

If Emby has been set up correctly, you'll be shown the setup wizard. Going forward, this is also how you'll administer Emby – through a web browser.

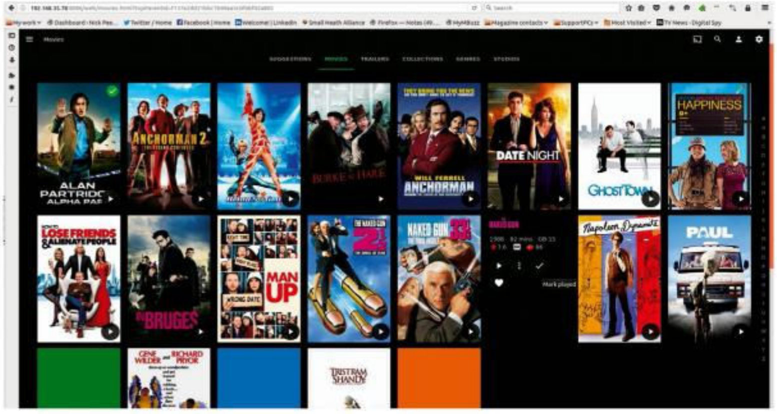
Next steps

One of the biggest issues with Emby is gaining access to your stored media. Start by creating a mount point:

```
$ sudo mkdir /media/emby
```

If your media is stored on an NTFS-formatted USB drive, you need to first add the following:

```
$ sudo apt-get install ntfs-3g
```



It'll take a little while, but eventually Emby will show off your media collection in all its glory.

AUDIOPHILE-FRIENDLY MUSIC

Imagine an inexpensive, pocket-sized audio player delivering superb quality digital sound through your home stereo or even a pair of headphones. No, we're not talking about your smartphone or tablet, but a standalone audio player streaming music from Spotify, SoundCloud, Google Music, web radio or local music files. That's the thought behind Pi MusicBox, a Pi-based project. You'll find all the details and the required software to download at the website, **www.pimusicbox.com**.

We paired a Pi Zero with the superlative Pi-DAC Zero (from £12, **www.iqaudio.co.uk**) and the Pi MusicBox distro to deliver a headless audio player based on Mopidy. Find the project at **www.mopidy.com**.

If you'd like to build a standalone audio receiver, then you can pair your Model A/B Pi with the Pi-DigiAMP+ (£55), power adapter (£22) and suitable case (£15.60). The Pi-DigiAMP+ delivers up to 35W stereo to a pair of two-wire speakers, which connect directly to its speaker terminals.

How about building an internet radio, or turning any speakers Wi-Fi-capable to play music from a connected USB device, NAS or mobile device? We tackled these two projects in the last issue of Pi User. If you missed it, visit **http://bit.ly/pi-user-01**

BUILD A PI ZERO STREAMING STICK

If you want to build a Raspberry Pi-powered media centre you can tuck behind the TV, you can pair a Pi Zero with the OSMC (**www.osmc.tv**) distro, which offers an optimised version of Kodi for lower-end machines. The initial setup requires access to a wireless hub to give you a wireless connection and USB keyboard/mouse for the initial configuration. (We suggest the Broadcom Adaptor, which provides a two-port hub and Wi-Fi from a single plug. Get it for £10 from most reputable Pi stores – see **www.modmypi.com**.)

Once set up, you can remove the adaptor and replace it with a smaller Wi-Fi adaptor – we're particularly fond of the Pi Zero Wi-Fi micro USB dongle from ModMyPi at £8, for example. While you can technically power your Pi Zero from your TV's USB port, this is not usually reliable – instead, use its own dedicated supply. If you want to free up your keyboard for some other project, you can control OSMC using any Kodi-compatible mobile app – for example try using Kore for Android.

For more help and advice, including some handy tips for streamlining a Kodi installation to improve performance on the Pi Zero, check out our guide on TechRadar at **http://bit.ly/kodi-on-pi-zero**

Now plug the drive into your Pi and type `sudo blkid` to obtain the UUID of your mounted drive. Write this down, then type `sudo /etc/fstab` to have it automatically mount at startup. Add the following line if it's a FAT32 drive, replacing the <UUID> with your drive's UUID:

```
UUID=<UUID> /media/emby vfat
iocharset=utf8,rw,umask=000 0 0
```

If it's an NTFS drive, use the following for read/write access:

```
UUID=<UUID> /media/emby ntfs defaults,uid=
1000,gid=100,dmask=027,fmask=137,utf8 0 0
```

You can also mount network drives too – so if your media is stored on an NAS drive, you can identify it using the **smbclient** tool as follows, replacing the "192.168.x.y" with your network drive's actual IP address:

```
sudo apt-get install smbclient
smbclient -L 192.168.x.y
```

Armed with this information, you can then populate the **fstab** file with something like the following, replacing "192.168.x.y" with your network drive's IP address, <sharename> with the folder you wish to access, and <username> and <passwd> with the username and password required to access the share in question:

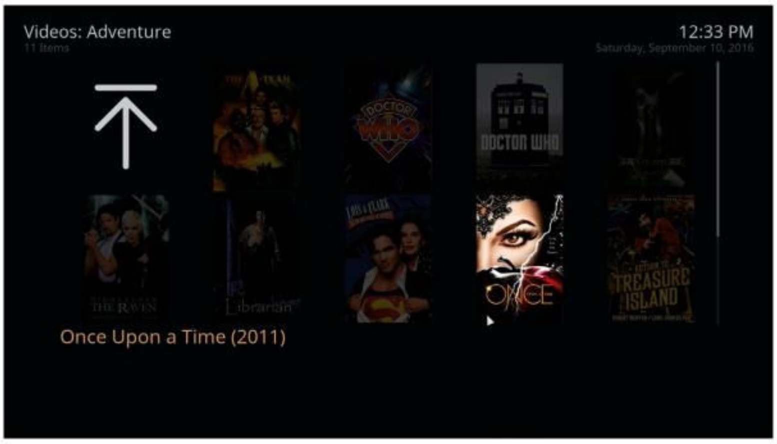
```
//192.168.x.y/<sharename> /mnt/<share1> cifs user
name=<username>,password=<passwd>,uid=1000,
gid=100 0 0
```

Save, exit and `umount -a` before `cd /media/emby` and `ls -l` to verify that the drive has been mounted and that your user has the required permissions to read and write data to and from it (`drwxr-x---` on USB drives, `drwxr-xr-x` on network drives).

Get set up

With your media drive now visible, it's time to get up and running using the Emby setup wizard in your browser. While stepping through this, avoid any processor-intensive options like "Enable chapter image extraction" when setting up your libraries. Also, if you're booting from a small (8GB to 16GB) SD card, we recommend ticking "Save artwork and metadata into media folders" to store all the data on your media drive.

You'll also be prompted to set up FFmpeg during the setup process – choose the "system installed version" when prompted, then click Next. From here it's all pretty straightforward and you'll soon find yourself at the Emby dashboard. Click the Home button in the top left-hand corner to browse your media. At first, you'll see colourful squares representing movie and other media, but over time, as Emby downloads metadata from the internet, these will be replaced with proper artwork as well as information about the shows, movies or music in question.



OSMC will happily run on a Pi Zero – and once you've done the initial configuration it can run headless.

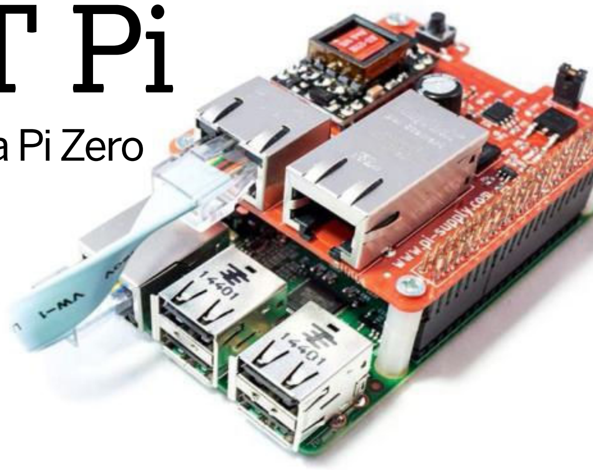
This can take some time to complete – this is the Raspberry Pi, after all – but you can immediately test your server by attempting to watch or listen to your library through the web browser. Click the Play button next to a title and wait for it to start – again there will be a short pause, but if all has been set up correctly, your media should begin to play correctly, indicating that it's now accessible.

From here you can explore the various ways of enjoying Emby-hosted media on other devices. Return to the Dashboard by clicking the menu in the top-left hand corner followed by Manage Server, and you can then set up other libraries, manage multi-user access and perform other tasks. For more about Emby Server and what it can do, see the tutorials and guides on the Emby Community site at **https://emby.media/community/index.php?/forum/49-server** – but note that some of the features covered, such as setting up live TV support, are probably beyond the capabilities of your Pi, and quite a few require some technical expertise to implement. We also recommend restricting the Raspberry Pi's use to purely serving locally hosted media that's been saved in a universal format (MP4 container, H.264/MP3 codecs) to save the Pi the job of having to try to transcode the stream.

»

The ultimate IoT Pi

Build a cheap and effective IoT device using a Pi Zero



The PoE Switch Hat powers your Pi using an Ethernet cable – but you'll need a suitable PoE injector or switch.

We're still waiting for a single Internet of Things standard to emerge, and chances are it never will. In the meantime, each time you purchase a piece of IoT kit from one manufacturer, be it a motion sensor, security camera or smart lighting, you're gambling that it will not only work with your current ecosystem but also prove flexible enough to work with others should the horse you're currently backing fall out of favour. And let's not talk about updates that break compatibility with older devices, eh SmartThings?

Services like **https://ifttt.com** go a long way to helping bridge the digital divide between devices, but why rely on a manufacturer's piece of expensive kit when you can custom-build your own versatile IoT gadgets using a Pi or Pi Zero? In this project, we're going to reveal the kit you need to build a core IoT device, then look at ways of adding to it and integrating it into various ecosystems.

Build your base unit

The key consideration for your IoT device is which model of Pi you want to build it around. The Pi Zero is on the surface the obvious choice thanks to its small size and low cost of entry, although you need to factor in additional effort and cost: in particular, you'll need to be confident using a soldering iron to add on much-needed functionality, such as the GPIO pins and even an alternative power source, as we'll see shortly.

At the other end of the scale, the Raspberry Pi 3 has the added advantages of built-in Wi-Fi and Bluetooth, and its extra processing power may be vital depending on how much functionality you want to pack into a single unit. It's also easier to add HATs including one that provides you with the means to power your Pi from its Ethernet port.

Next, consider your device's main purpose. If you want to use it as a camera, for example, then ask yourself whether you want a camera for daytime or night-time use – pick the Pi-NoIR model for the latter, paired with an infrared light source. If you plan to use the camera with your Pi Zero, remember to purchase the required cable (an extra £4). Also take the time to examine what cases are out there – two in particular caught our attention. The ZeroView (£7, **thepihut.com**) is a clever case and mount for the Pi Zero that attaches to glass surfaces, such as a window, and provides you with the means to monitor what's happening on the other side with minimal glare or reflection. Or look at the Pi Camera Box Bundle (£24, **www.modmypi.com**), which provides a case, wide-angle lens and wall mount for Pi B+, 2 and 3.

Another potential use for your IoT device is as a sensor. The two obvious choices here are the SenseHAT (£25, from various outlets) for the main

BUILD YOUR OWN SOFTWARE PROJECTS

Get started programming your IoT device in Wyliodrin STUDIO by using the Visual Programming component.

Okay, so you've set up your hardware, but how do you go about integrating these new devices into your IoT infrastructure? While extensively documented and supported, your hardware usually requires some Python programming skills to make use of its extensive libraries. If you're just starting out with programming or feel that you want more flexibility in this kind of project, you should take a look at the open-source Wyliodrin IDE.

Wyliodrin is a web-based platform that tries to simplify the programming process by supporting a wide range of languages in a platform-agnostic way to let you put your projects together using a more user-friendly building-block approach called "Visual Programming" while the code is written for you in the background.

You'll need to start by signing up for a free account at **www.wyliodrin.com**, then input the basic details of your Pi – its type and network connection (wired or wireless). Next, download the pre-built Raspbian image to a suitable microSD card – you can also add Wyliodrin to an existing installation via a lengthy script, but it takes hours to complete. Once this is done, embed a custom .JSON file that links your Raspberry Pi to your Wyliodrin account and reboot. As soon as you're logged in, your Pi should appear online in your Wyliodrin web console, ready for you to program.

From here the simplest approach is to install the Wyliodrin STUDIO Chromium extension. This extension supports JavaScript and Python, but it also gives you Visual Programming as an accessible way into coding, and the dashboard helps to visualise what you're actually doing. Of course, if you're already a programming whizz you can easily switch to code view if you prefer.

Raspberry Pi and the recently released Enviro pHAT (£16, Pimoroni), sized to match the Pi Zero and offering temperature/pressure, light level/colour, motion and analogue sensors (but you'll need to solder the supplied headers to it). If you're looking for a straightforward motion sensor, then the PIR Infrared Motion Sensor (£3, ModMyPi) is all you need – it's also built into the SPi-Box (£13, **shop.sb-components.co.uk**) case, ready for pairing with your Pi camera.

Power your device

A critical consideration for your IoT device is powering it. It's not always possible to find a convenient wall socket, so you may be tempted to go down the portable power route. The problem here is that you'll need to recharge the batteries on a daily basis in most cases, so if your Pi is supposed to be left to its own devices 24-7 then this is not a practical solution.

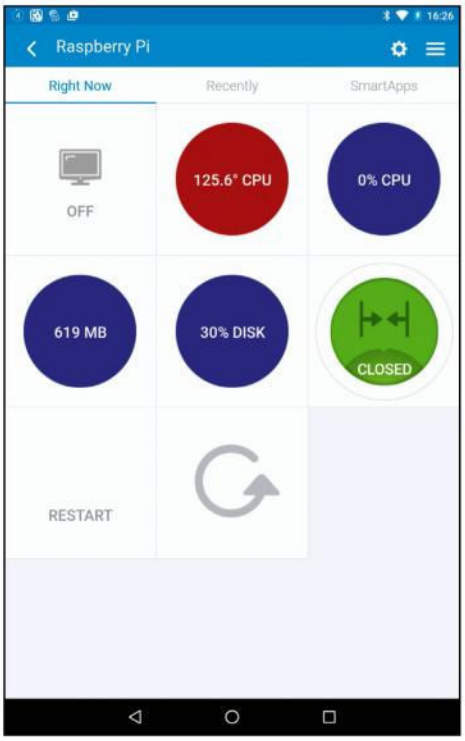
One often overlooked option is Power Over Ethernet. Ethernet can be transmitted up to 100 metres, and the cables can carry both power and network, so you don't have to bother with a Wi-Fi adaptor or worry about weak or dead signals. A 100m cable can cost as little as £20, but spending a bit more will get you cable rated for outdoor use – perfect if you've mounted your Pi at the other end of the garden in a waterproof location. Just one critical consideration before you begin: you'll need a switch or router that provides 802.3af PoE support, or buy a special 802.3af PoE injector that sits between your switch/router and your Pi. A good injector is TP-Link's TL-POE150S (£20, PC World), which provides its own power supply and connects to any free port on your switch or router. You then plug the other end into the PoE switch on your Pi, which can then draw up to 13W of power.

If you plan to power multiple devices then a full-blown PoE-compatible switch may seem attractive, given that you often get up to four ports offering PoE connections. But check the total power budget before you begin: for example TRENDnet's TPE-S44 switch costs under £35 from Ebuyer.com and offers eight 10/100Mbps ports in total (four of which support PoE), but it can only supply a total 30W across those ports. That should be sufficient for low-powered headless devices, but doesn't give you much room for manoeuvre.

If this means of power appeals, then the PoE Switch Hat (£30, **pi-supply.com**) will work with the Pi B+ or later, and it features a handy programmable power switch too. A Model B-compatible switch can be purchased for £61 (**www.xtronix.co.uk/raspberry-pi-poe.htm**).

If you want to use PoE with the Pi Zero then you'll need to go down the DIY route. One project in active development is the PoEPi: Pi Zero project, which has the added benefit of also providing your Pi Zero with an Ethernet port. It's shaped to sit neatly on top of the Pi Zero board, but it's still early days. You can follow its progress at **https://hackaday.io/project/9455**.

You'll find a working guide to building your own rudimentary PoE switch for the Pi Zero at **http://n-o-d-e.net/tagged/howto** – find the entry for 11/01 marked 'How to power a Raspberry Pi Zero over Ethernet'. You'll need to source a USB Ethernet adaptor that works with the Pi Zero, plus 12V-5V step-down convertor, two micro USB plugs and thin wires. It's a seven-step process, which requires a soldering iron, clippers, wire strippers and needle nose pliers, but there's a handy YouTube video provided to give you a visual guide to follow.



You'll need to set up BerryIO in order to incorporate your Pi into the SmartThings ecosystem.

Integrate with other IoT kit

If you're looking to integrate your Pi into an existing IoT ecosystem but starting from scratch then the obvious choice is Energenie, thanks to its close integration with the Pi. On top of its own hub, sensors and other IoT-related gadgets, there's the Pi-mote Control, with which you can remotely switch electrical sockets on and off using your Pi, plus a Mi|Home Adapter Plus, which monitors the electrical consumption of an appliance attached to an Energenie smart power socket.

It's also possible to integrate your Pi into other ecosystems. Visit **https://github.com/nfarina/homebridge** for a guide to emulating Apple's HomeKit API on your Pi, for example. Samsung's SmartThings API is another popular choice, and if you go to **https://github.com/nicholaswilde/berryio-smartthings** you'll find a way to bridge the two using BerryIO as an intermediary. There's a fair bit of work involved, and references to "My Device Type" in the SmartThings setup process actually refer to the "My Device Handlers" section. Nevertheless, it works, giving you remote control of your Pi through the SmartThings app among other things.

WHY RELY ON MANUFACTURERS' EXPENSIVE KIT WHEN YOU CAN CUSTOM-BUILD YOUR OWN?

If all this feels like too much effort, there's one other way you can use your Pi with other devices – through **IFTTT.com**, which enables you to combine devices and services using a series of triggers and actions in the form of recipes that easily cross the technology divide. Almost any IoT gadget worth its salt is represented here, and you basically call scripts from your Pi using the Maker channel to control or interact with other devices. ■

Mac FORMAT

macOS Sierra is here...

iOS10
Essential advice in every issue

LET THE EXPERTS AT MACFORMAT BE YOUR GUIDES TO APPLE'S POWERFUL NEW OS

Available from www.macformat.com

Beginners guide to RetroPie
hulu
NETFLIX
Amazon
HBO GO
HBO NOW
SHO ANYTIME
Open Database - kp3.kdbx
Enter Master Key
Master Password:
Key File:
Windows User Account
Help
By default this script
the IP address
"My IP address is %s\n" "$_IP"
/home/pi/stockticker.py &

The Top 20 Raspberry Pi issues

Things can go wrong with even the best technology. Here's how to tackle some of the most common glitches you're likely to encounter with your Pi

The Raspberry Pi is elegant in its simplicity. This may explain why in a blog post in 2013 the Raspberry Pi Foundation revealed that only 0.02 percent of Pis are returned as defective. Of these around half were actually experiencing an issue that could easily be solved – so easily in fact that 10 year old Jessica, the daughter of Director of Software Engineering Gordon Hollingworth, was able to separate the good Pi units from the genuinely faulty ones.

This being the case, if you are experiencing issues with your Pi, please don't throw it away in frustration or send it back with an irate letter. It's very likely that your issue can be solved (if not by a 10-year-old, then using a remedy found by one of the other Pi users who've experienced it).

Over the next 10 pages we'll explore some of the more common problems with the Raspberry Pi and provide some tips on how to resolve these with minimal fuss.

Pi problems typically revolve around a power adapter simply not providing enough current for your Pi and connected devices, corrupted microSD cards, and out-of-date software. More specific issues can sometimes occur with keyboards, displays, sound and internet connectivity.

Do not despair, however, if these solutions fail to resolve your problem. The Raspberry Pi online community are extremely active and willing to donate their expertise free of charge. Many of the tips outlined here focus on isolating your issue. Where possible try to cone down to precisely where the problem lies. For instance, if your Pi won't connect to your home wireless network, try connecting at the office. If your camera module fails to switch on, use a different power adapter, and so on.

Above all, the best tip to begin with is to make sure that any hardware you buy, including the Pi itself and all cabling, comes from an authorised distributor. Pi kit is not expensive (just compare the prices of Apple branded tech, for example!), and pinching those few pennies is usually a false economy. So, let's look at the 20 most common Pi problems and how to overcome them…

»

Pi continuously reboots

Pi unable to boot up? Check your power supply and SD card

1 One of the most fundamental issues with the Pi may be that it fails to boot up properly at all. Usually the Raspberry Pi logo will appear, then the screen will transform into a block of rainbow colour as the Pi self-tests its graphics processor. Then occasionally the Pi will either display scrolling text on a black screen or simply reboot repeatedly and fail to load the OS.

The most common reason for this is that your power supply is simply not providing enough juice. The Pi is powered by a 5V micro-USB cable. The amount of current required will vary depending on your model. For the latest Raspberry Pi 3 Model B, use a 2,500mA supply to make sure the Pi switches on and that all four USB ports are powered. For a detailed list of the Pi's power requirements visit **www.raspberrypi.org/help/faqs/#powerReqs**.

Speaking of USB ports, bear in mind that the power requirements for the Pi will vary depending on what devices are attached. The Pi will usually display a lightning bolt symbol at the top right of the desktop if it is underpowered but can also simply reboot if too much current is being drawn.

Ideally, you should buy an official Raspberry Pi power supply from an authorised dealer such as the Pi Hut. Where possible, try to connect the Pi directly to the adapter itself rather than through a USB hub.

Although insufficient power is the most common reason that a Pi is stuck in a boot cycle, there are other reasons why this may occur. If your SD card has been formatted incorrectly or an update to Raspbian has gone awry, the Pi may also reboot. Try installing NOOBS to your microSD card and reinstall Raspbian if changing the power supply doesn't help. See **www.raspberrypi.org/documentation/installation/noobs.md** for help with this.

This Micro USB 5V Supply is available for just £5 from the Pi Hut (**https://thepihut.com/products/micro-usb-power-supply-for-the-raspberry-pi**)

No HDMI monitor

No problem, just use composite video

2 Fortunately for users with older televisions, the Raspberry Pi 3 has a combined 3.5mm audio/composite video jack, which means even TVs without an HDMI port can be used as a display. You will need a 3.5mm to RCA A/V Composite cable, available from sites such as ModmyPi.com. Whichever supplier you use, make sure that the A/V composite cable has three RCA connectors, so both audio and video are supported.

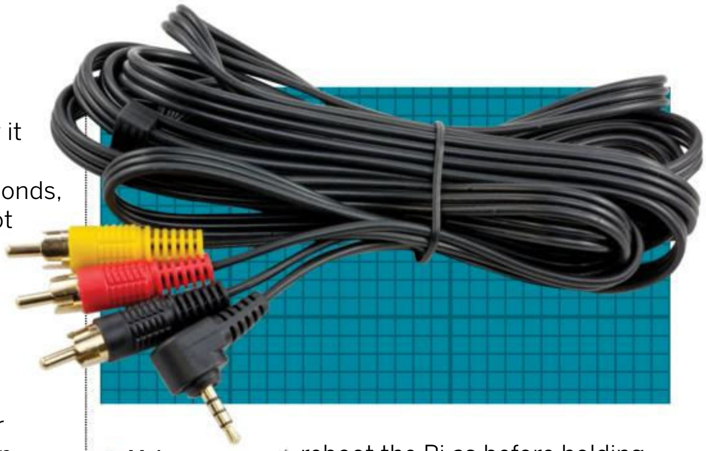
If you have a standard installation of Raspbian, then the Pi should automatically switch to using your composite video when it doesn't detect an HDMI device. If you are using NOOBS to install Raspbian, however, by default it outputs only to HDMI. You can modify this during startup. First connect your composite cable and keyboard to your Pi. Power it on while holding down the Shift key for around 10 seconds, then release the key. Do not worry at this stage if you don't see anything.

To switch the Pi from HDMI to the composite display, press number 3 on your keyboard for PAL televisions or number 4 for NTSC. Televisions bought in the UK and Ireland will almost certainly use PAL, whereas those in North America will probably use NTSC.

If the display appears distorted or black-and-white, most likely you have selected the wrong standard. Reboot the Pi and repeat the steps above to try another format.

Should you later decide to invest in an HDMI television, reboot the Pi as before holding Shift, then hold down the number 1 key to switch back to HDMI.

In the unlikely event your TV uses the Japanese version of NTSC or the Brazilian version of PAL, or if you wish to change the aspect ratio used by the Pi, you'll need to modify the Pi's config.txt file accordingly. See **http://elinux.org/RPiconfig#Video** for information on this.


Make sure your composite cable has three RCA connectors, like this one from ModMyPi (www.modmypi.com/raspberry-pi/accessories/audio-cables/av-composite-cable-3.5mm-to-3-x-rca-3m).

Pi sticks on rainbow screen

Don't panic if the Pi fails to boot beyond the GPU test loading screen

3 We mentioned opposite that on booting up, the Pi runs its own GPU (Graphics Processing Unit) self-test, which appears as a dazzling rainbow coloured screen. Occasionally the Pi may hang on this screen and refuse to load the OS.

This can happen when the Raspbian kernel image is corrupted. An easy way to test this is to use your computer to install Raspbian onto a new microSD card, insert this new card into the Pi and try to boot. See **www.raspberrypi.org/documentation/installation/installing-images** for more information on this.

While this solution works well in theory, it can be a nuisance to have to configure your Pi all over again. You might also have personal files on the original microSD card. If so, first back up your data: power off, remove the Pi's SD card, insert it into a card reader, then connect this to a computer. The card's second partition should contain your **/home/** folder. Make a copy of this to a safe place such as your computer's desktop.

To try to repair Raspbian's kernel, you must run a system update. If the Pi will not boot normally, it's best to try to connect via SSH. If you haven't used SSH before, see problem 7 over the page. For recent versions of the Raspbian OS, to enable access this way you need to save a file named **ssh** onto the SD card's **/boot** partition.

If the Pi fails to load beyond this screen, power off and insert the microSD card into your computer.

If you're able to connect to your Pi via SSH, run the following two commands to attempt to reinstall the Pi kernel:

```
sudo apt-get update
sudo apt-get install raspberrypi-kernel
```

Once the update has completed, also edit the **config** file with the following:

```
sudo nano /boot/config.txt
```

Add the text **boot_delay=1** on a new line. Press Ctrl + X, then Y, then Return to save and exit.

Now try to boot the Pi again. If the OS still won't load, you'll need to run a fresh install of Raspbian on another SD card as outlined above.

Keyboard types wrong letters

Internationalise your Pi using the Configuration file

4 As the Raspberry Pi was invented in the UK, the default keyboard layout is British English. This can cause some headaches for users with US or International keyboards. Even when keyboards are manufactured in the UK, yours may have a different layout or extra keys. Fortunately the Pi can be configured accordingly.

Open Terminal on your Pi or connect via SSH and run the following command:

```
sudo raspi-config
```

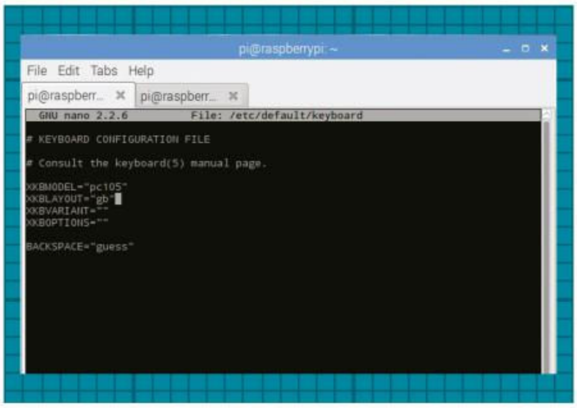
Use the arrow keys to choose Option 4 (Localisation Options) and press Enter. Next choose Option 3 (Change Keyboard Layout). This will allow you to choose your specific model of keyboard, such as "Apple". Use the arrow keys to click OK to confirm your changes and save the modified file.

Although it's not essential for mapping keys, you may also wish to change your system locale, language and time zone to your own country. In the Localisation menu select option 1 to list various locales and languages. These use the two-letter country code (see below). You can of course change this back to English at any time if you wish.

On the main menu, you can also choose option 2 to change your time zone. Raspi-Config will first ask you to choose your general geographic area, such as Europe, before honing down to specific countries.

If you have a foreign keyboard, usually selecting the correct locale will map the correct keys to the Pi. If not, run the following command to open the keyboard configuration file:

```
sudo nano /etc/default/keyboard
```



```
  GNU nano 2.2.6          File: /etc/default/keyboard

# KEYBOARD CONFIGURATION FILE

# Consult the keyboard(5) manual page.

XKBMODEL="pc105"
XKBLAYOUT="gb"
XKBVARIANT=""
XKBOPTIONS=""

BACKSPACE="guess"
```

Foreign keyboard? Change "gb" to the correct two-letter country code for your keyboard.

Find the line reading **XKBLAYOUT="gb"** and replace **"gb"** with the two-letter code of your keyboard country, such as **"us"**. You can find a full list of the two-letter country codes at **www.worldatlas.com/aatlas/ctycodes.htm**.

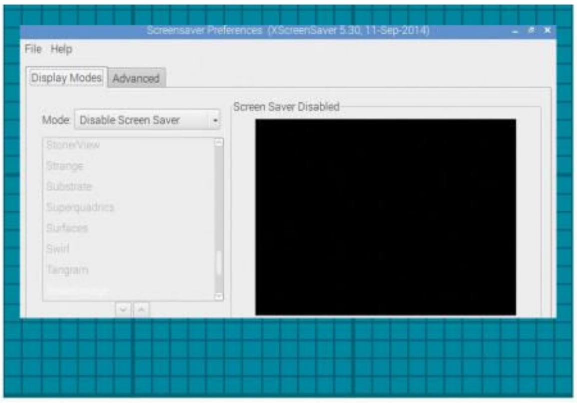
Press Ctrl+X, then Y, then Return to save and exit.

»

Screen goes blank periodically

Disable the screensaver to keep your display running at all times

5 The Raspberry Pi is designed to draw as little power as possible. For this reason, by default the screen will blank after around ten minutes of mouse and keyboard inactivity. This can be a huge liability when you're undertaking various Pi projects, particularly where the Pi is being used to display the time or weather.



Mode: Disable Screen Saver

Screen Saver Disabled

Click the Mode menu, then Disable Screensaver to stop the screen display powering off for good.

If you've connected the Pi to a touch screen, it's usually not too much trouble to tap it and bring it back to life. However, if you are using a regular display without a keyboard or mouse, the only way to wake the screen up is to connect a keyboard or restart the Pi altogether.

Various well-meaning online contributors in Raspberry Pi forums have suggested some command line hacking which involves extensive use of the Pi terminal and altering various values, which is not ideal for newcomers to the Pi. You can in fact disable the screen saver very easily from the Pi by installing the screen saver management program xscreensaver. Open Terminal on your Pi and run the following two commands:

```
sudo apt-get update
sudo apt-get install xscreensaver
```

The program itself takes only around 11MB. Once installation is complete, click the Pi Menu and go to Preferences. You will see there is a new option there named Screensaver. Click this, then click the drop-down menu labelled Mode, then Disable Screensaver in the main window.

If you wish simply to use an actual animated screen saver rather than disable it altogether, note that the author of xscreensaver doesn't recommend using this rather outdated version of the program. When using xscreensaver you may even see an alert to this effect. Consider installing a more recent screen saver utility such as gnome-screensaver if you want something more colourful. To install this, simply run:

```
sudo apt-get update
sudo apt-get install gnome-
screensaver
```

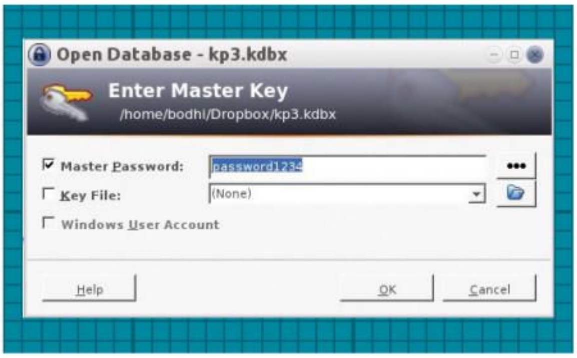
Password problems

What is my password? Why can't I see it?

6 Some newcomers to the Raspberry Pi may be surprised that no letters appear when typing passwords, particularly in the Terminal program. This is a feature, not a bug. Raspbian is based on Debian Linux, and Linux is extremely security conscious, so it doesn't allow user passwords to be displayed. On the plus side, if this is a concern for you, you needn't worry – Raspbian is behaving exactly as it should.

The default user password is set to "raspberry" on first run, which isn't too tricky to type. If you do find you're having trouble with this, first check that you have the correct keyboard layout (see problem 4 on the previous page). If this fails to resolve the issue, go to Applications > Accessories > Text Editor and type your admin password there to make sure it's written correctly. Next in the upper menu click Edit > Select All, then Edit > Copy. You can now paste the password into the Pi terminal when prompted by holding Ctrl + Shift + V.

For multiple and/or lengthy passwords, consider using a password manager such as Keepass. Programs like Keepass store all of your passwords in an encrypted database. This database is protected by a "master key" – a single password which can be input to retrieve the others. Password managers such as Keepass also allow you to right-click entries in the database to copy and paste the password into fields. This can save an enormous amount of time and trouble.



☑ Master Password: password1234

☐ Key File: (None)

☐ Windows User Account

Click the three dots next to the master password field to see what you are typing. You can do the same for other entries in the password database.

To install Keepass, go to Applications > Preferences > Add/Remove Software and enter "Keepass2" in the search bar. Tick the first entry in the window labelled "Password Manager" and then click Apply at the bottom right.

Once it's installed, you can then launch Keepass from Applications > Accessories. If you need help getting started visit the Keepass Forum at **https://forum.keepassx.org**.

Trouble connecting via SSH

Enable SSH to connect to your Pi remotely and run commands without a monitor

7 SSH (Secure Shell) is a set of network protocols for securing connections between computers. In plain English this means that you can use a utility on another device such as a PC to control your Pi, pretty much as if you were using the Pi's Terminal directly. Visit **www.raspberrypi.org/documentation/remote-access/ssh** for information on connecting to the Pi via SSH from different platforms such as Mac (which has SSH built-in) or Windows (which needs it added).

Because your Pi has a default username and password, it could be targeted remotely by a hacker, and for this reason SSH is disabled by default. If you can connect your Pi to a monitor temporarily, you can enable SSH yourself. First open Terminal and run the following command:

```
sudo raspi-config
```

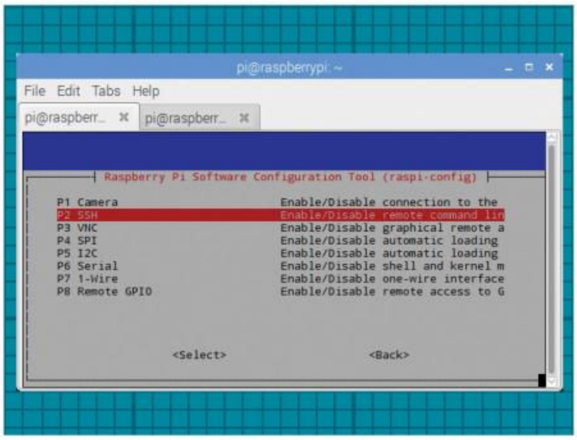
Using your arrow keys, choose Option 5 (Interfacing Options), then select it by pressing Enter. Next choose Option P2 (SSH). Raspi-config will ask if you're sure you want to enable. Press Enter once again to confirm. You will see a confirmation message stating "The SSH server is enabled".

If you do not have a monitor, don't worry. Power off the Pi, remove its SD card, insert it into a card reader, and plug this into your computer. Find the **/boot** partition and create a file there named **ssh**. You can use a text editor such as Microsoft's Notepad or Apple's TextEdit to do this but make sure that the file has no filename extension such as .txt. Safely eject the SD card and reinsert into your Pi. SSH will now be enabled.

The next time you load the Pi with a monitor attached, you will see a warning that the default password for the Pi user has not been changed from "raspberry". To fix this, open Terminal and enter the following command:

```
sudo passwd pi
```

You'll be asked to enter the new password twice. Note that this command itself can also be run over SSH.



```
┤ Raspberry Pi Software Configuration Tool (raspi-config) ├

P1 Camera                      Enable/Disable connection to the
P2 SSH                         Enable/Disable remote command lin
P3 VNC                         Enable/Disable graphical remote a
P4 SPI                         Enable/Disable automatic loading
P5 I2C                         Enable/Disable automatic loading
P6 Serial                      Enable/Disable shell and kernel m
P7 1-Wire                      Enable/Disable one-wire interface
P8 Remote GPIO                 Enable/Disable remote access to G

           <Select>                      <Back>
```

If possible, hook your Pi up to a monitor to enable SSH using raspi-config.

Touchscreen troubles

Overcome the most common quibbles with the Raspberry Pi touchscreen

8 The official Raspberry Pi 7-inch touchscreen display is an incredible add-on. When combined with a battery pack and an on-screen keyboard, it can even transform your Pi into a mobile device.

The touchscreen is designed so that the Pi can be connected via the DSI (Display Serial Interface) cable and screwed onto the back of the display. Power is supplied by jumper wires which connect to the GPIO pins on the Pi so that it and the display can share power from the same microUSB cable.

Connecting the devices requires a little time and patience. If you buy the display via the Pi Hut website, there is a link to assembly instructions (**https://thepihut.com/blogs/raspberry-pi-tutorials/45295044-raspberry-pi-7-touch-screen-assembly-guide**)

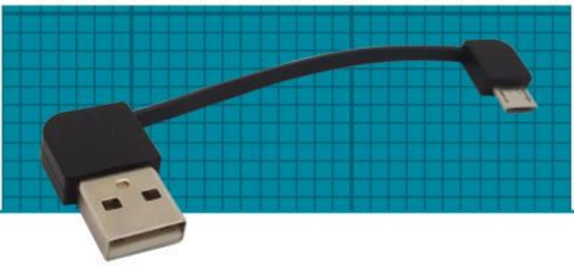
Before attempting to assemble the screen, provide yourself with a flat, well-lit space. Make sure to run updates on the Pi to be certain you have the latest version of Raspbian too:

```
sudo apt-get update
sudo apt-get upgrade
```

The display comes with four M.25 screws, so buy yourself a suitable screwdriver. Make sure the display powers on correctly before connecting anything permanently. Be extremely careful when pulling out the small clamps which attach to the DSI ribbon – gently pull straight backwards with finger and thumb. The entire apparatus is fairly delicate and, in keeping with the "bare bones" feel of the Pi, there's no back case to the display, although you can buy one separately if you wish.

The default setup can cause issues if you want to use the GPIO pins for projects. The Pi Hut has an innovative solution in the form of a Short MicroUSB Power Cable which neatly attaches to the USB port, no messy wires required.

This handy cable gives you use of the GPIO pins when using the touchscreen display. See http://bit.ly/pihut-microUSB for information.



If you wish to change the orientation of the touchscreen display, open Terminal and run the following:

```
sudo nano /boot/config.txt
```

Paste the text **lcd_rotate=2** in a new line at the very bottom. Save and exit. The display will reorient the next time you reboot.

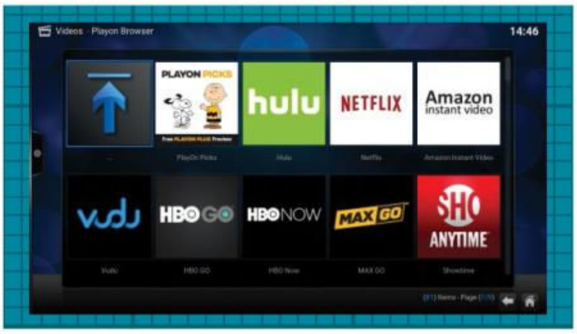
»

Playing Flash or DRM content

Can't play DRM protected content? Streaming video is easy... if you don't use Flash

9 In September 2016, Adobe announced that it will once again develop its Flash Player for Linux, having previously abandoned the project in 2016. This doesn't do much good for Raspberry Pi users, though, because the player will not work with the Pi's ARM processor architecture.

There are unofficial plugins for the Pi's Chromium browser such as Pepperflash which can theoretically be copied from another machine to work with the Pi, but you need to be very comfortable with Linux and use of the command line to do this. Instead, bear in mind that major sites such as YouTube and Daily Motion now stream video via HTML5, which the Pi supports. Flash is generally not required.



Kodi Media player on the Pi can stream Netflix and Hulu from a Windows PC via PlayOn.

The Pi is designed to be free and open source. As such it does not contain proprietary software for playing content encrypted by various forms of DRM (Digital Rights Management) and streamed via providers such as Netflix and Hulu. One possible workaround if you have a Windows PC in addition to your Pi is to install PlayON (available from **www.playon.tv**). This program enables you to stream Netflix content from the PC to other devices such as a Raspberry Pi running Kodi media player. See the Kodi Player Wiki (**http://kodi.wiki/view/HOW-TO:Install_Kodi_on_Raspberry_Pi**) for installation instructions.

If you are using Raspbian with NOOBS, you can also simply restart the Pi holding Shift to install the LibreELEC Operating System, which is specifically designed to display Kodi content.

If you wish to watch DVD or Blu-ray discs, either via an external DVD drive or from a "ripped" ISO image, you can also purchase an MPEG-2 and/or VC-1 Licence key inexpensively from **www.raspberrypi.com/license-keys**. In order to place the order, you'll need to provide your Pi's serial number, which you can find by opening Terminal on your Pi or connecting via SSH (see problem 7) and running the following command:

```
cat /proc/cpuinfo
```

Big SD cards not recognised

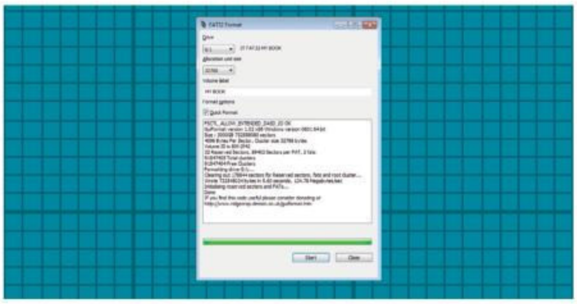
Format large capacity SD cards to be recognised by the Pi

10 Since the Pi was first launched, the cost of microSD cards with a capacity of 64GB and more has fallen, allowing for much greater amounts of data to be stored. If however you use the official SD Card Formatter tool on the SD Association's website (**www.sdcard.org/downloads/formatter_4**) to format high-capacity microSD cards, the Pi might not recognise them. This is because the tool uses the exFAT file format for cards over 32GB in capacity, and this is not recognised by the Pi.

In order to make sure your Pi will recognise your high-capacity card, first format it using the official tool. Next use a third-party tool to format the SD card's partition to FAT32, which is recognised by the Pi. For Windows, the Raspberry Pi Foundation recommends using Guiformat (**www.ridgecrop.demon.co.uk/guiformat.htm**). Mac and Linux users can use their OS's own disk utility to format the entire card to FAT32.

Once this is done, continue with installation of NOOBS onto your microSD card by following the steps at **www.raspberrypi.org/documentation/installation/noobs.md**.

Card still not recognised? First visit the Pi Foundation's website for a list of other programs you can use to format a card (**www.raspberrypi.org/documentation/installation/sdxc_formatting.md**). If all else fails, digital cameras use a slightly different method for formatting microSD cards, so you could try inserting the card and formatting it using your camera's built-in firmware.



Windows users can use the Guiformat tool to prepare the SD card. Mac users can use the built-in Disk Utility in macOS.

Another easy workaround for Raspberry Pi 3 users is simply to boot from a USB stick or hard drive instead of an SD card. Note, though, that this feature is still experimental. You will need to enable USB boot mode on the Pi, as well as format the USB device in a specific way. For instructions, see the Raspberry Pi website at **www.raspberrypi.org/documentation/hardware/raspberrypi/bootmodes/msd.md**.

Sound problems

Troubleshoot sound issues and select the correct speakers for output

11 If your Raspberry Pi is plugged into an HDMI television, sound will be output there and should work out-of-the-box. If not, you will need to isolate the issue to find out where the sound problem lies. In the first instance, eliminate the obvious by clicking the volume control during playback to make sure the volume is turned up and is not muted.

If this fails to resolve the issue, find out whether this issue is related to playing only certain content by using Terminal to play some of the WAV files bundled with the Pi. These files are saved in ***/usr/share/scratch/Media/Sounds***. For instance, run the following command:

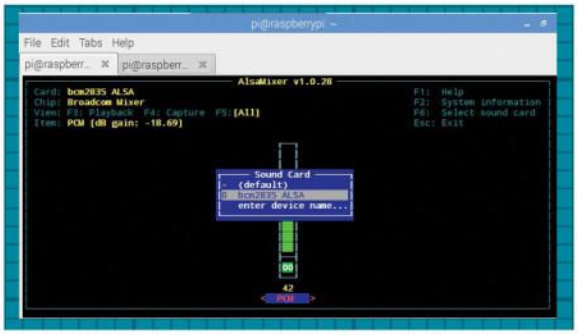
```
aplay /usr/share/scratch/Media/
Sounds/Vocals/Singer2.wav
```

If you hear sound, the issue is likely to be related to media codecs. Go to Applications > Preferences > Add/Remove Software and install VLC Media Player. This should enable your Pi to play almost all audio and video formats.

If the issue persists, then double-check your Pi's sound settings by opening Terminal again and running the command:

```
sudo alsamixer
```

This retro interface allows you to alter the volume of your various sound devices (by default there is just one) and even select another sound card if you have one. By default only playback devices are displayed but you can press F5 to include sound recording devices if you so wish.



```
AlsaMixer v1.0.28
Card: bcm2835 ALSA                              F1:  Help
Chip: Broadcom Mixer                            F2:  System information
View: F3: Playback  F4: Capture  F5:[All]       F6:  Select sound card
Item: PCM [dB gain: -18.69]                     Esc: Exit

                    Sound Card
                 -  (default)
                 0  bcm2835 ALSA
                    enter device name...

                        00
                        42
                       PCM
```

Use the up/down keys to control your device's volume. Press F6 to choose a different sound card.

If selecting the correct sound device and increasing the volume fails to fix the issue, try resetting your sound settings with the following command:

```
sudo /etc/init.d/alsa-utils reset
```

Certain computer monitors connected by HDMI may not output audio correctly. You also may want to connect speakers or headphones to the Pi's 3.5mm jack. If so, first open Terminal and run the following:

```
sudo raspi-config
```

Choose Option 8 (Advanced Options), then Audio. This will enable you to switch the output between HDMI and the headphone jack manually.

USB devices not recognised

Eliminate some common pitfalls of USB devices

12 The Pi supports a wide range of USB flash drives and other devices, and it's usually a matter of simply plugging and playing. If however nothing happens when you first insert the device, try to hone down on the actual issue.

Your first step should be to try the device with another non-Pi computer such as a desktop PC to check whether it's simply faulty. If it is, contact the seller or manufacturer about a replacement under warranty.

If it's not faulty, open Terminal on your Pi and run the following command:

```
sudo dmesg -C
```

Next insert your USB device and run the command

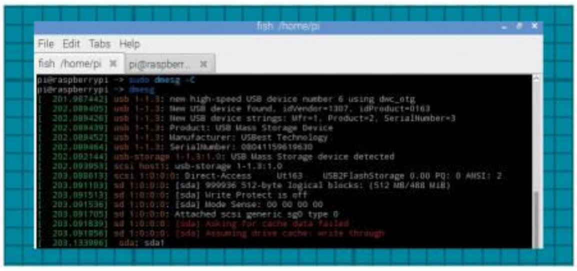
```
dmesg
```

If your USB device has been detected, then it will be listed here. You should also see any error messages relating to it.

If the USB device is not listed then there's either a hardware fault or it possibly requires more power than the Pi's USB ports are able to supply. Consider buying a powered USB hub from a Pi reseller.

If the device is detected but still is not working, it's possible that the Pi doesn't have the software drivers necessary to interface with your USB device. To begin with, use Terminal to run the following commands to ensure you have the most recent version of Raspbian:

```
sudo apt-get update
sudo apt-get upgrade
```



```
pi@raspberrypi ~> sudo dmesg -C
pi@raspberrypi ~> dmesg
[  201.987442] usb 1-1.3: new high-speed USB device number 6 using dwc_otg
[  202.088405] usb 1-1.3: New USB device found, idVendor=1307, idProduct=0163
[  202.088426] usb 1-1.3: New USB device strings: Mfr=1, Product=2, SerialNumber=3
[  202.088439] usb 1-1.3: Product: USB Mass Storage Device
[  202.088452] usb 1-1.3: Manufacturer: USBest Technology
[  202.088464] usb 1-1.3: SerialNumber: 08041159619630
[  202.092144] usb-storage 1-1.3:1.0: USB Mass Storage device detected
[  202.093953] scsi host1: usb-storage 1-1.3:1.0
[  203.088613] scsi 1:0:0:0: Direct-Access     Ut163    USB2FlashStorage 0.00 PQ: 0 ANSI: 2
[  203.091103] sd 1:0:0:0: [sda] 999936 512-byte logical blocks: (512 MB/488 MiB)
[  203.091513] sd 1:0:0:0: [sda] Write Protect is off
[  203.091536] sd 1:0:0:0: [sda] Mode Sense: 00 00 00 00
[  203.091705] sd 1:0:0:0: Attached scsi generic sg0 type 0
[  203.091839] sd 1:0:0:0: [sda] Asking for cache data failed
[  203.091856] sd 1:0:0:0: [sda] Assuming drive cache: write through
[  203.133986]  sda: sda1
```

Use dmesg to check for any USB device error messages. Here a flash drive has been detected but was not unmounted properly.

If updating your system fails to resolve the issue, contact the device manufacturer in the first instance to ask if there is a suitable Linux driver. If the manufacturer says there isn't or you cannot contact them, don't despair. There may be a community-maintained driver developed independently of the manufacturer or the driver for a similar device may work.

With the USB device still inserted run the following command in the Pi Terminal to list all attached USB devices:

```
lsusb
```

Note down the device's ID, e.g. 1307:0163, and enter it into your favourite search engine with the words "Linux driver".

»

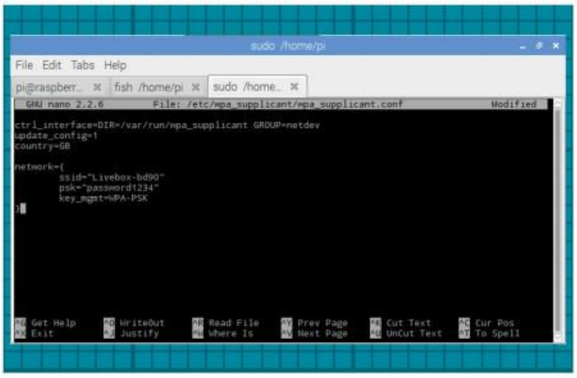
Wi-Fi connection issues

Troubleshoot issues with connecting to Wi-Fi networks

13 If you're using the Pi 3, then Wi-Fi is built-in. If you're using a separate USB wireless stick, follow the steps we've outlined under problem 12 on the previous page to make sure the stick itself isn't the problem.

Certain routers have an upper limit on the number of devices which can be connected to a network. Try connecting a new device such as a mobile phone to the same wireless network to check if you have the same difficulty. If possible, try to set up a hotspot on your phone and attempt to connect to that from the Pi to further identify where the issue lies.

If your Wi-Fi network is not listed, enter the settings here manually. Use raspi-config to change the country if the code isn't correct.



```
  GNU nano 2.2.6          File: /etc/wpa_supplicant/wpa_supplicant.conf          Modified

ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
update_config=1
country=GB

network={
        ssid="Livebox-bd90"
        psk="password1234"
        key_mgmt=WPA-PSK
}

^G Get Help    ^O WriteOut    ^R Read File   ^Y Prev Page   ^K Cut Text    ^C Cur Pos
^X Exit        ^J Justify     ^W Where Is    ^V Next Page   ^U UnCut Text  ^T To Spell
```

If you are abroad, different countries have varying regulations on what Wi-Fi channels can be used. You can configure the Pi to use these by opening Terminal and running

```
sudo raspi-config
```

Choose Localisation Options, then Change Wi-Fi Country. You will see a list of countries. Select your current location using your arrow keys.

The issue may be security. By default the Pi 3 cannot connect to any wireless network secured by WEP (Wireless Encryption Protocol). Contact the router manufacturer for help to change the wireless security protocol to WPA (Wi-Fi Protected Access) or ideally WPA2.

If this fails to resolve the issue, open Terminal on the Pi or connect via SSH and run the following command:

```
sudo nano /etc/wpa_supplicant/
wpa_supplicant.conf
```

This contains the name, settings and password for all Wi-Fi networks connected to the Pi. If your network is listed, verify that the password and other settings are correct. If not add your network at the bottom of the file in the following format:

```
network={
    ssid="your-network-ssid-
name"
    psk="your-network-password"
}
```

If you have an SD card reader this is also an excellent way to connect your Pi to your Wi-Fi network without using a monitor.

RetroPie gaming problems

Solve common problems with RetroPie through using the GitHub page

14 One of the most common uses for the Pi is as a retro games emulator. If like many others you wish to enjoy playing arcade classics of the '80s and '90s, your first port of call should be RetroPie's GitHub page (**http://bit.ly/RetroPie-setup**).

Some users buy Pi bundles on sites like eBay which include the Pi itself, an SD card with RetroPie pre-installed and some pre-installed games, known in retro-gamer parlance as "ROMs". If the microSD card doesn't work out of the box, go through the RetroPie setup again to make sure you have the latest version.

The latest version of RetroPie should automatically detect and configure most controllers connected via USB. If you want to set a wireless Bluetooth controller, follow the instructions for your specific gamepad on the GitHub page. For instance steps for the PS3 Controller can be found at: **http://bit.ly/RetroPie-PS3**. The Raspberry Pi 3 has built-in support for Bluetooth and will provide the best performance. If you're using other Pi models you will need a Bluetooth USB dongle to use wireless controllers.

For copyright reasons, RetroPie does not come with any ROMs. If you have a legal copy of a ROM, the easiest method to transfer it to the Pi is to create a folder named **retropie** on a USB stick, then create another named **roms** inside it. Safely eject the USB stick from your computer and insert it into the Pi. RetroPie will automatically create subfolders for various games consoles. You can then reinsert the USB stick into the computer and download the ROMs to there.

MENU
NEXT
Beginners guide to
RetroPie

Before using RetroPie make sure to read the "First Installation" page on GitHub, which should resolve most common issues.

The Internet Archive (**http://archive.org**) maintains large amounts of "abandonware" ROMs, which they claim are free to download. Make sure this is legal in your jurisdiction before copying to RetroPie.

For troubleshooting some common sound, video and speed issues, visit the RetroPie wiki at **https://github.com/retropie/retropie-setup/wiki**.

Display and desktop glitches

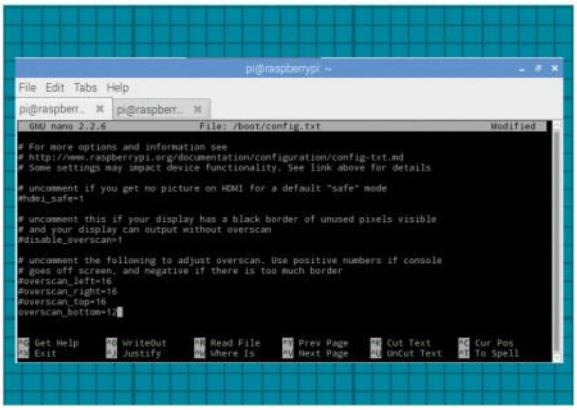
Border and resolution problems, including overly large desktops, are easily solved

15 By default the Pi's display resolution is 800x400 pixels. This may well not cause any issues but certain monitors are set up for a higher resolution. On startup the Pi will request a monitor's settings and if the Pi cannot find a suitable one, parts of the desktop may not display, or the display may seem blurry, or possibly both.

If you see a black border of unused pixels, try to disable overscan on the Pi. Open Terminal or connect via SSH and run the command

```
sudo nano /boot/config.txt
```

Before making any changes, make a note of the original settings in your config.txt file. That way if the display becomes unreadable, you'll be able to connect via SSH or plug your microSD card into a card reader and restore the original settings.



```
  GNU nano 2.2.6            File: /boot/config.txt                     Modified

# For more options and information see
# http://www.raspberrypi.org/documentation/configuration/config-txt.md
# Some settings may impact device functionality. See link above for details

# uncomment if you get no picture on HDMI for a default "safe" mode
#hdmi_safe=1

# uncomment this if your display has a black border of unused pixels visible
# and your display can output without overscan
#disable_overscan=1

# uncomment the following to adjust overscan. Use positive numbers if console
# goes off screen, and negative if there is too much border
#overscan_left=16
#overscan_right=16
#overscan_top=16
overscan_bottom=12

^G Get Help    ^O WriteOut    ^R Read File   ^Y Prev Page   ^K Cut Text    ^C Cur Pos
^X Exit        ^J Justify     ^W Where Is    ^V Next Page   ^U UnCut Text  ^T To Spell
```

Make a careful note of any lines you modify to make sure you can revert to previous settings.

Now find the line reading **#disable_overscan=1** and use the backspace key to remove the # at the start of the line. Press Ctrl + X, then Y, then Return to save and exit. Reboot the Pi to have the changes take effect.

If you have the opposite problem in that parts of the desktop are not visible (i.e., it extends past what the monitor can display), then instead of disabling overscan, remove the # at the start of the lines **overscan_left**, **overscan_right**, **overscan_top** and **overscan_bottom**, and manually adjust the values there, using negative numbers if necessary to reduce the border size.

If the entire desktop displays but is overly large or small, or the above steps fail to resolve the issue, you may need to adjust your console size. Disable overscan as outlined above. Next scroll down to the section beginning **uncomment to force a console size**. Remove the # symbols at the start of the two lines beginning **framebuffer** to set an alternative resolution. For instance if the current height and width are 1280x720 you may wish to adjust this to 1024x768. Reboot to apply your changes.

Vanquishing VNC

Master the art of connecting remotely to your Pi's desktop

16 VNC (Virtual Network Computing) is a way to control one computer remotely from another. This is especially handy in the case of the Pi as it allows you to use your computer keyboard and mouse. The most recent versions of Raspbian actually come with VNC Server software provided by RealVNC pre-installed. This makes accessing the Pi much easier, but for security this handy feature is disabled by default. In order to begin using it open Terminal on your Pi or connect via SSH and run the command

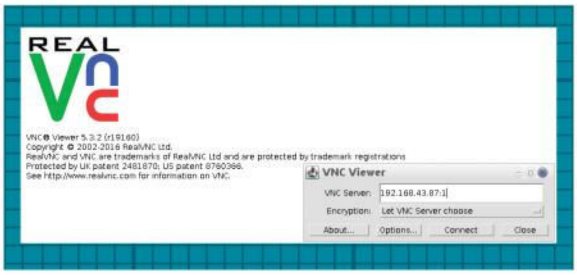
```
sudo raspi-config
```

Go to Option 5 (Interfacing Options), then select VNC. Hit Enter while Yes is highlighted and VNC will be enabled.

If you already have a monitor attached to your Pi, double-click on the colourful VNC icon to see the connection details and the server signature, which is used to verify that no-one is trying to spy on your connection.

If you have a web browser that supports Java on your computer, simply navigate to the address listed, e.g. **http://192.168.4.87:5800**. You will see a warning message the first time you connect. Choose to continue and enter your Pi's current username and password. The desktop will appear in a new window and you can access it just as on the Pi. If you don't have Java-enabled browser or prefer to access the Pi from a mobile device, visit **www.realvnc.com** for links to suitable software.

If you're running your Pi "headlessly" – without a monitor – then connect via SSH (see problem 7, page 39) and enable VNC by running the command



VNC® Viewer 5.3.2 (r19160)
Copyright © 2002-2016 RealVNC Ltd.
RealVNC and VNC are trademarks of RealVNC Ltd and are protected by trademark registrations
Protected by UK patent 2481870; US patent 8760366.
See http://www.realvnc.com for information on VNC.

VNC Server: 192.168.43.87:1
Encryption: Let VNC Server choose

To connect to the virtual desktop, make sure to enter the entire IP address, including the colon and number e.g ":1".

```
vncserver
```

This will start the VNC software. You'll see a message stating the IP address of the virtual desktop, for example:

```
"New desktop is raspberrypi:1
(192.168.43.87:1)"
```

Make a note of this. Now in your Java-enabled web browser go to **http://ipaddressofyourpi:5800**. In the VNC Viewer enter the IP address you noted to connect to the virtual desktop.

»

Programs don't auto-run

Ensure your software starts up without prompting

17 Certain Pi projects, particularly those where the Pi is continuously displaying information like the weather or stock prices, work much better if they launch automatically. For instance if the Pi reboots after a power cut, it can be very annoying to have to start the various programs up all over again manually. This is very easy to do. To make a program launch at startup, first open Terminal on your Pi or connect via SSH and run the command:

```
sudo nano /etc/rc.local
```

Find the line reading "**exit 0**". It's essential to place your commands *above* this line. For example, if you have a Python script, you could paste the following above "exit 0":

```
python /home/pi/stockticker.py &
```

The ampersand at the end of the line tells the Pi to continue booting once it has run the script. Press Ctrl + X, then Y, then Enter to exit and save your changes.

Some projects may require you to run a script at certain times of day. For instance you may have a script that automatically checks the temperature and sends you an e-mail summary. You can schedule this using the tool Cron. Open the Terminal and run the following command to open Cron:

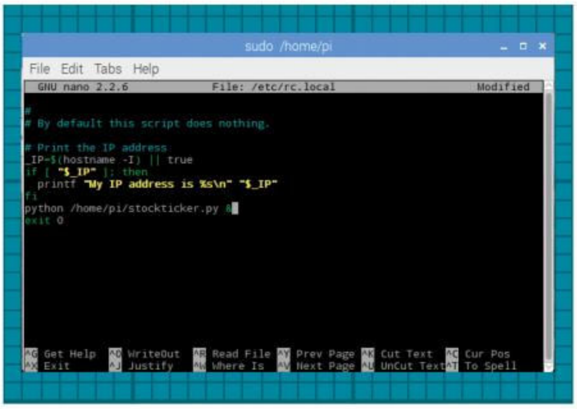
```
crontab -e
```

Press Return to choose the default Editor. Scroll down to the very bottom of the file to add your scheduled job. The basic format for Cron is **minutes/hours/day of month/month/day of week**. Naturally you can customise this however you want. For instance to schedule the script "check-temperature.sh" to run every hour, you'd add:

```
0 * * * *  /home/pi/check-temperature.sh
```

Save and exit in the usual way. If you're having trouble working out how to write the scheduled job in the timing format used by Cron, the helpful website **https://crontab.guru** allows you to enter numbers to be sure you have it exactly right.

This script will launch automatically on boot. Make sure to put any commands above "exit 0".



```
  GNU nano 2.2.6              File: /etc/rc.local                    Modified

#
# By default this script does nothing.

# Print the IP address
_IP=$(hostname -I) || true
if [ "$_IP" ]; then
  printf "My IP address is %s\n" "$_IP"
fi
python /home/pi/stockticker.py &
exit 0

^G Get Help   ^O WriteOut   ^R Read File  ^Y Prev Page  ^K Cut Text   ^C Cur Pos
^X Exit       ^J Justify    ^W Where Is   ^V Next Page  ^U UnCut Text ^T To Spell
```

Camera calamities

Iron out issues with official Raspberry Pi Camera Module

18 The Pi's Camera Module V2 is designed to be easy to set up and use, and is compatible with the Raspberry Pi 1, 2 and 3. The camera attaches via a ribbon cable and usually works out-of-the-box.

The Raspberry Pi Foundation website (**www.raspberrypi.org/documentation/configuration/camera.md**) cautions that the Pi Camera Module is very sensitive to static. So it's worth investing in an earthing strap to ground yourself before touching either the camera or the Pi. Be extremely gentle when pulling out the tabs on the end of each connector to attach the ribbon cable. Gently push the tabs back on the camera end to lock the cable in place or you may find it comes loose. You should hear a slight click when the connector is correctly in place.

Once the camera is connected open Terminal on your Pi and run

```
sudo raspi-config
```

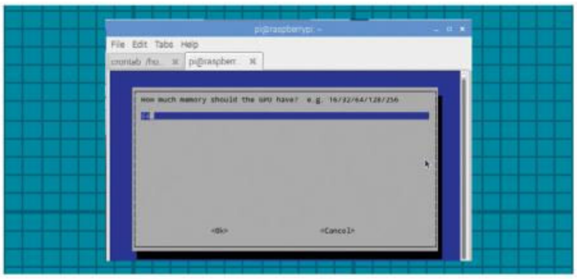
Scroll to Option 5 (Interfacing Options) and then press Enter over the first option (Camera). Choose Yes to enable it.

If video is not working after you've enabled the camera, make sure you have connected the camera ribbon to the CSI-2 (Camera Serial Interface), not the DSI (Display Serial Interface). Make sure also that the ribbon connectors are seated straight in their sockets.

If the camera is enabled and connected correctly, try running the following commands to ensure Raspbian is up to date:

```
sudo apt-get update
sudo apt-get upgrade
```

If the camera is still not working, double-check your power supply. The camera draws around 250mA of extra current. See our solution to problem 1 (on page 36) for tips to ensure you're using a reliable adapter.



```
How much memory should the GPU have?  e.g. 16/32/64/128/256
```

Assign at least 128MB of memory to the GPU to be certain the Pi can handle footage from the camera.

Should the camera module still fail to load, try assigning more memory to the GPU. This allows the Pi to handle video better. Open Terminal and run

```
sudo raspi-config
```

Choose option 7 (Advanced Options), then Option A3 (Memory Split) to set this.

Unavailable software

Unable to install new programs, or unsure what programs to use? All is not lost...

19 Although the Pi no longer has a dedicated app store, clicking Applications > Preferences > Add/Remove Software will open a window which enables you to choose from the various programs available in the Pi's software repositories.

Categories are available on the left, and you can click on individual programs to see a more detailed description before installing. Sometimes, however, the Add/Remove Software window may hang or say it is unable to find a package.

If this happens it's most likely because your software sources are out of date. Click Options at the top left of the Add/Remove Software window and choose Check for Updates. Alternatively open Terminal on your Pi or connect via SSH and run:

```
sudo apt-get update
```

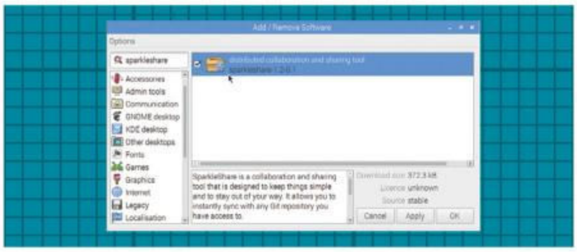
Ideally you should do this before attempting to install any new applications. It can't hurt to run the following as well to make sure you have the most recent version of the Add/Remove Software program:

```
sudo apt-get upgrade
```

If you're uncertain whether a program is available for the Pi, use the search bar in the Add/Remove Software window or run the following command (be sure to spell the program correctly):

```
sudo apt-cache search
<softwarename>
```

If the software seems to be unavailable for the Pi, use your favourite search engine to see if there's a suitable alternative. Not all programs will work on the Pi, even if there's a Linux version, because the Pi has an ARM processor rather than Intel, like most desktop machines. Check the developer's website first to see if there is an ARM version of the software available. If there isn't but the developer has provided the source code of the program, you can attempt compiling it yourself on the Pi. This is tricky to get right, can be time-consuming, and usually requires additional software libraries as well as the source code. Visit **www.aboutdebian.com/compile.htm** for a basic introduction to transforming raw code into a workable program. Use the most recent version of the source code unless you're specifically told otherwise.



SparkleShare is a collaboration and sharing tool that is designed to keep things simple and to stay out of your way. It allows you to instantly sync with any Git repository you have access to.

Case in point: The official Dropbox client doesn't run on the Pi but the open source alternative Sparkleshare is compatible.

When all else fails...

Take some time to research your issue before giving up

20 While we've covered some of the more common problems with the Pi, there is a near infinite number of ways in which yours can be set up, so there's no way to cover every eventuality. Whatever your problem, however, there are some standard troubleshooting approaches you can try. Even if you're not very experienced with computers, try to eliminate likely or possible causes before concluding that the Pi must be physically faulty or that you have done something wrong.

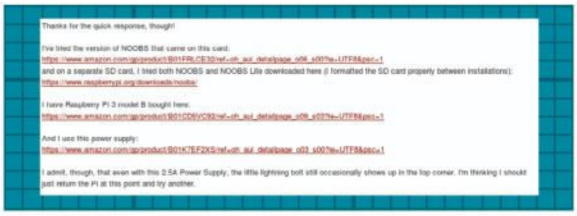
Does the problem occur when you use a different power adapter? Does running an update or trying using a new microSD card make any difference? If you use wireless peripherals such as a keyboard and mouse, does the same issue occur with a cabled device?

If you conclude that the problem relates specifically to the Pi itself, then consider posting on the Raspberry Pi forums (**www.raspberrypi.org/forums**) to ask for help. Do this even if you see that someone has posted about a similar issue, because they may have used an older version of the Pi and/or Raspbian. Your first post must be pre-approved by a Moderator, so do not be surprised if you don't see it right away.

Try to include as much information as possible in your post. If you are receiving long error messages, use PasteBin (**http://pastebin.com**) to copy and paste the text, then provide the link to this in your post.

If your problem relates to using one specific program and the Pi otherwise works properly, see if you can contact the developer or find a user forum for that program. The same holds true for hardware devices, although the Raspberry Pi forums may be able to point you in the direction of the correct software driver (see above).

If after troubleshooting you find that your Pi falls into the 0.01 percent of devices which actually are faulty, contact your Pi distributor to arrange a replacement or a refund. Do not contact the Raspberry Pi Foundation – it does not sell the devices itself.



Thanks for the quick response, though!

I've tried the version of NOOBS that came on this card:
https://www.amazon.com/gp/product/B01FRLCE32/ref=oh_aui_detailpage_o06_s00?ie=UTF8&psc=1
and on a separate SD card, I tried both NOOBS and NOOBS Lite downloaded here (I formatted the SD card properly between installations):
https://www.raspberrypi.org/downloads/noobs/

I have Raspberry Pi 3 model B bought here:
https://www.amazon.com/gp/product/B01CD5VC92/ref=oh_aui_detailpage_o08_s07?ie=UTF8&psc=1

And I use this power supply:
https://www.amazon.com/gp/product/B01K7EF2XS/ref=oh_aui_detailpage_o03_s00?ie=UTF8&psc=1

I admit, though, that even with this 2.5A Power Supply, the little lightning bolt still occasionally shows up in the top corner. I'm thinking I should just return the Pi at this point and try another.

Providing links to any accessories you purchased enables other users to check the exact model number.

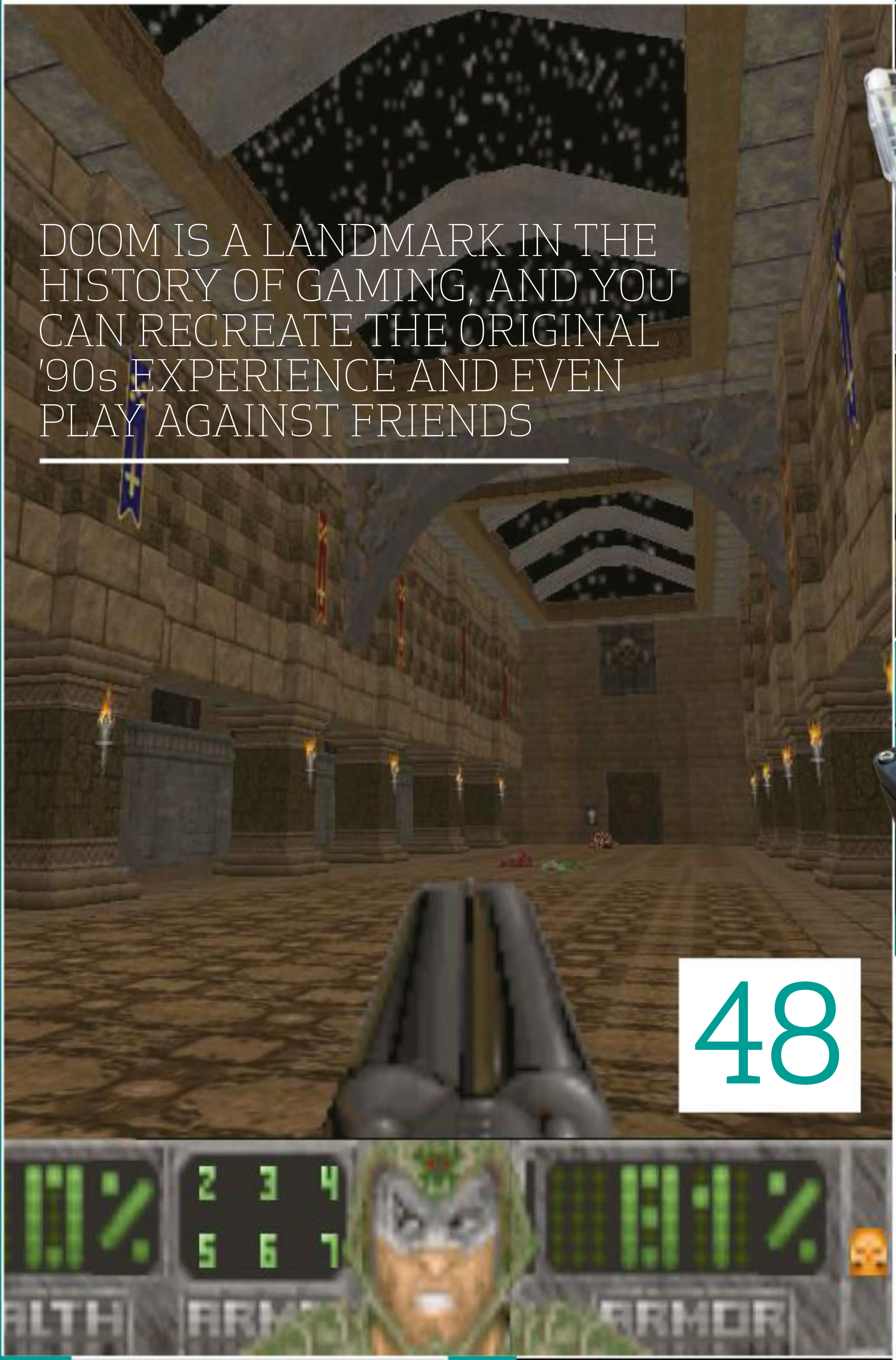
THE EASY WAY TO
LEARN WINDOWS

CLEAN YOUR PC
THE COMPLETE GUIDE TO REINSTALLING WINDOWS
FREE TO PLAY
THE BEST FREE WINDOWS GAMES TO TRY TODAY!
NEW SECTION! WELCOME TO WINDOWS 10
Windows Help & Advice
100 WAYS TO IMPROVE WINDOWS 10
Even faster and smarter ways to do everything!
Customise your Start menu
Use keyboard shortcuts
Edit and share photos
Use virtual desktops
BEGINNERS' GUIDE
Set up Edge, the brand new Windows 10 browser
EXPERT GUIDES 50 PAGES OF WINDOWS HELP!
100% JARGON FREE
NEW MINI PC REVIEWED!
PLUS: Windows 10 laptops • High-res music players • New Sony mini projector & more!
Future
CHRISTMAS 2015

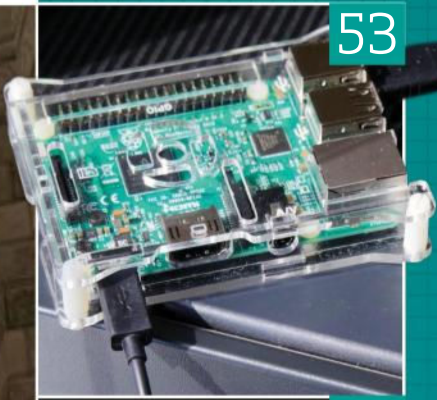
AVAILABLE IN STORE AND ONLINE
www.myfavouritemagazines.co.uk

Pi PROJECTS

Put your Pi to use – we show you how, step-by-step!

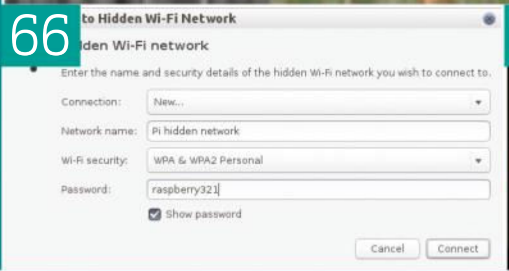


DOOM IS A LANDMARK IN THE HISTORY OF GAMING, AND YOU CAN RECREATE THE ORIGINAL '90s EXPERIENCE AND EVEN PLAY AGAINST FRIENDS



56
KODI
Mashup Zone
Cute Kittens And Funny Kitten Videos Compilation 2016
PICTURES
VIDEOS

61
talkiepi


66

to Hidden Wi-Fi Network

den Wi-Fi network

Enter the name and security details of the hidden Wi-Fi network you wish to connect to.

Connection: New...

Network name: Pi hidden network

Wi-Fi security: WPA & WPA2 Personal

Password: raspberry321

☑ Show password

Cancel
Connect

71
[3D images]
[Cute Devil]
[Fruits]
[Natural Scenery]
[Sunsets]
[Things for sale]
SALE

48 Play Doom on your Pi

53 Make any printer wireless

56 Create a Pi Media Centre

61 TalkiePi: Make a walkie-talkie

66 Protect yourself with a firewall

71 Build a Pi-based photo gallery

DOOM Shareware - Chocolate Doom 2.2.1
YOU GOT THE SHOTGUN!
200
50
50
300

› Love it or hate it, Doom redefined the gaming industry.

Play Doom on your Raspberry Pi

Nate Drake adds a touch of chocolate to your Raspberry Pi through installing a version of this iconic First Person Shooter

Doom is a landmark in the history of gaming, set in the not too distant future. You are a Space Marine, exiled to Mars for a dull, routine assignment after striking a superior officer. The Martian Marine Base is secretly conducting experiments into teleportation through creating wormholes between Mars' two moons.

Events take a horrifically dramatic turn for the worse when one of the moons, Deimos, suddenly disappears. Taking the role of the hero of this cyber-tragedy, you are quickly dispatched on a shuttle to the other moon, Phobos. You soon discover that the hapless research scientists have accidentally opened a doorway to hell itself and that you will have to fight your way through demonspawn, possessed personnel and the forces of darkness to have any hope of escaping.

Despite the retrospectively clichéd plot and clunky graphics, id Software's Doom has enjoyed huge popularity since it exploded onto users' PCs in 1993. It helped cement the First Person Shooter (FPS) genre of video games as well as spawning a number of official sequels, spin-off games and fan-made levels.

Even those who are too young to remember when Doom was initially released will likely be familiar with the FPS genre through the numerous examples around. As Doom is the most iconic, it's important to recreate the original '90s experience as closely as possible and for this reason we have selected the "Chocolate Doom" Engine to run on your Raspberry Pi.

This project assumes you have a Raspberry Pi 2 or 3, running the latest version of Raspbian. Use the commands `sudo apt-get update` then `sudo apt-get upgrade` to be sure of this. If you want to play Doom's multiplayer Deathmatch mode, you will also need your Pi to be connected to your home network and/or the internet.

Getting the Lowdoom

Being a First Person Shooter means that all the action is experienced as if you're looking through the eyes of the main character (unnamed in Doom but sometimes referred to as "Doomguy").

The basic premise of the original game is that you must fight through the moons of Phobos, Deimos and then Hell itself. This is accomplished through punching, shooting and even (if you wish) gashing with a chainsaw any zombies, flaming skulls and demonic henchmen that get in your way as you progress through all 27 levels.

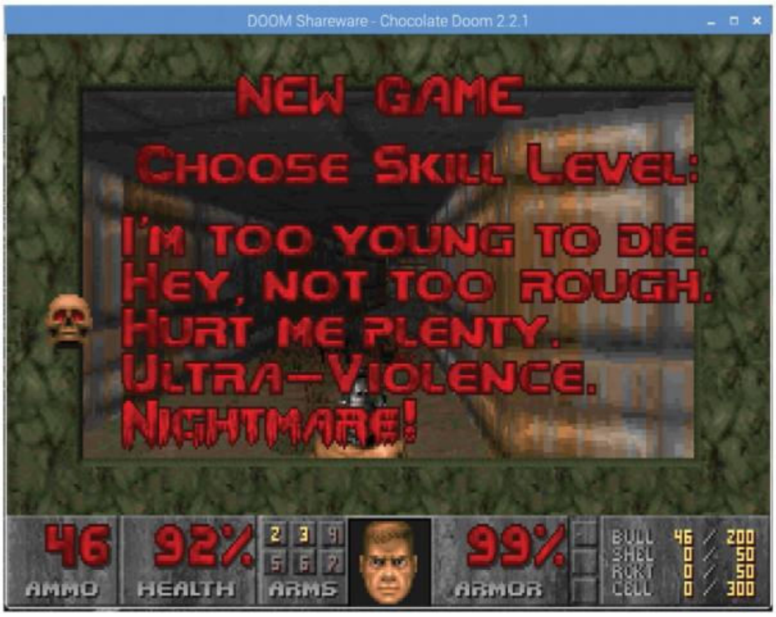
The satanic imagery and ultra-violence were not welcomed by several religious groups and political pressure groups, who campaigned in vain to either censor Doom or make it more family friendly after its release. Germany banned the game out of hand for containing "bloody, sadistic violence", although this ban was repealed in 2011.

However, there appears to be no such thing as bad publicity. Doom sold 1.1 million copies between 1993 and 1999, making it the eighth best selling game of that time. Developer id Software claimed it was probably the number one cause of reduced productivity globally. This was meant as a joke but by 1995 there were an estimated 10 million PCs globally with a version of Doom installed.

Gameplay is simple in theory if not in practice. On each level you must locate the exit to the next area. Helpfully enough these are marked with an exit sign. However, you have to make your way through a maze of levels, identical corridors and locked doors. Some help is available in the form of power-ups, which have been liberally scattered around the place by persons unknown. These include key cards, armour, weaponry and first aid kits. You can also make use of the in-game map by hitting the Tab key to see where you have been.

Monsters have an irritating habit of showing up when least expected. They also guard valuable items and exits and have an unfortunate tendency to shoot at you or throw fireballs from a distance.

Although Hell has numbers on its side, the enemy AI is not particularly advanced and they will attack each other if one gets in another's way.



On start-up you can choose from various imaginatively-named difficulty levels.

Another advantage is the arsenal of in-game weapons. Although you begin with a pistol, as the game progresses you can lay your hands on shotguns, chain-guns, plasma rifles and even the unholy BFG (Big Fragging Gun) 9000.

Chocolate and Vanilla

Both "Chocolate Doom" and "Vanilla Doom" are different forms of the Doom Engine. It helps to think of the Doom Engine as been the executable parts of the game, as opposed to the WAD files (see "Where's all the Data" overleaf) containing raw data like graphics, sound, weapons and so on. The engine handles aspects of the game like the overall structure of levels and movement. The official Doom Engine developed by id is known as "Vanilla" because of its relatively bland flavour. »

Pi bites

A reboot of Doom was released in May 2016 by id Software for Microsoft Windows, PlayStation 4 and Xbox. We live in hopes there'll be a Pi version in due course.

CHEATERS NEVER PROSPER

Chocolate Doom mirrors the original closely, including the built-in cheat codes. These can be used at any time during play, simply by typing them on the keyboard. In most cases you will see a message confirming that the cheat has been activated. The cheats below assume you're using a QWERTY keyboard.

Cheats can work in multiplayer Deathmatches, depending on the settings made by the person setting up the game – it is possible to add the flag "**– nocheats**" to prevent others from having an unfair advantage.

All cheats are prefixed by "id", the name of the original game's developer.

A few select cheats in no particular order are:

idchoppers: Why shoot the spawn of Hell humanely when you can carve them up with a gardening implement? This cheat will equip you with the chainsaw. Press 1 to select it.

idclev##: Use this cheat code to warp to the Episode or Map of your choice – for example **idclev02** would take you to map 2. This has the same effect as starting a new game in that your health will return to 100 percent and any power-ups and/or weapons you picked up will be lost.

iddqd: This cheat sets your health permanently to 100 percent. This "God Mode" will make you immune to most forms of damage, though certain types of floors can still harm you. You can also be killed by "telefragging" – where other players or creatures teleport onto a spot where you're standing.

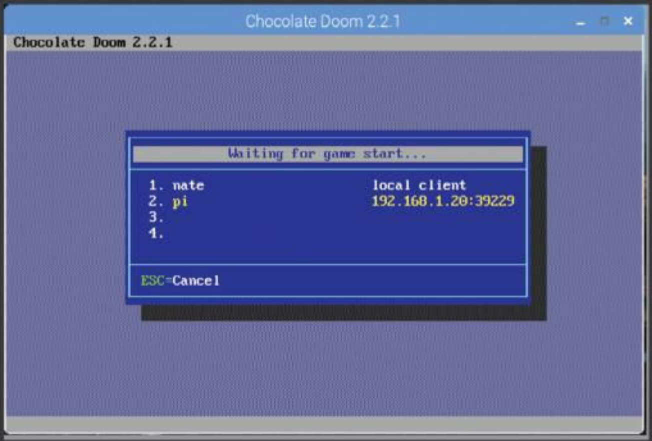
DOOM Shareware - Chocolate Doom 2.2.1
100% HEALTH
ARMS
0% ARMOR
AMMO

Enabling God Mode leaves you free to get up close and personal with the chainsaw.

There are plenty more along these lines for the lazy, desperate or perhaps less skilled player. For a full list of cheats see the Doom Wiki at **http://doom.wikia.com/wiki/Doom_cheat_codes**.

DOOM DEATHMATCHES

Chocolate Doom supports multiplayer with minimal fuss. Unlike "Vanilla Doom", you can play Chocolate Doom either over the internet or via LAN (local area network). To keep things simple, however, make sure all the players who wish to take part are running Chocolate Doom and using the same WAD file before going ahead.



```
Chocolate Doom 2.2.1

              Waiting for game start...
  1. nate                        local client
  2. pi                          192.168.1.20:39229
  3.
  4.

 ESC=Cancel
```

When all players have joined, press the space bar to begin.

Whether you play multiplayer over the internet or via a LAN, one of the participating players must act as the server and host the game. Rest assured that the host player will be able to play the game too. UDP port 2342 must be open on each player's router and/or firewall. (See "The Doom is in the Details" on the next page for more info.)

To host a Deathmatch game on your Pi, open Terminal or connect via SSH and run the following command:

```
chocolate-doom -server -privateserver -deathmatch
```

In order to join your game, internet players should run the following command:

```
chocolate-doom -connect 1.2.3.4
```

Replace 1.2.3.4 with the public IP address of the host player's machine. You can find your public IP by visiting websites such as **http://whatismyip.com**.

For players who are connected to a local network, life is even easier: simply run the game in the usual way but add the argument **-autojoin**, for instance:

```
chocolate-doom -WAD DOOM1.WAD -window 640x480 -autojoin
```

The multiplayer window will be displayed, showing the various clients which are connected. Press the space bar when ready to start playing.

Further help for Multiplayer Mode, if you need it, is available from Chocolate Doom's Wiki: **www.chocolate-doom.org/wiki/index.php/Multiplayer**

Pi bites

Budding coders can view the original Doom Source code at **https://github.com/id-Software/DOOM**

Doom Enthusiasts refer to "Vanilla Doom" to distinguish between the official game engine for running Doom and any variants. The distinction is important because the Doom Engine was modified and re-used in several commercial games such as Heretic and Hexen.

Since id released the source code of the Doom Engine in 1997, a number of community developers have also adapted it to their purposes. Chocolate Doom is one such project.

Despite modifying the game engine, Chocolate Doom aims to reproduce the experience of playing the original version of Doom as much as possible. Any modifications to the original game engine do not interfere with the authentic gaming experience. More importantly Chocolate Doom strives to be compatible with all Doom Expansion files (WADs) that were designed to work with Vanilla Doom.

TeamTNT's Eternal Doom was praised for its rich graphics, stunning architecture and huge levels.

AMMO
HEALTH
ARMS
ARMOR

The only area where Chocolate Doom differs significantly from the original is when playing in the multiplayer Deathmatch mode. One player's machine has to act as a server, unlike in Vanilla Doom where players would connect directly to each other. Fortunately this is very easy to set up (see *Doom Deathmatches* above).

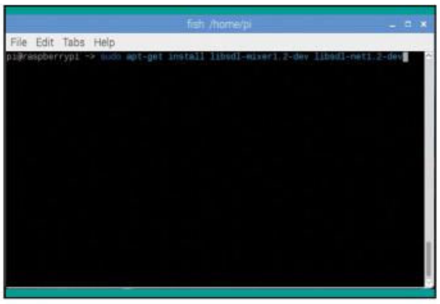
Where's all the Data?

To play Doom you'll need both the engine and some so-called WAD files. WAD (Where's All the Data) files contain game data such as details of levels, graphics, sound effects and background music. The idea behind WADs is to make it easy for people to make custom levels and modifications for the game. WADS exist separately from game engines such as Chocolate Doom and must be downloaded from elsewhere. There are two main types of WADS. For the purposes of this project we'll focus on iWADs (short for internal WADs), which contain data for entire levels.

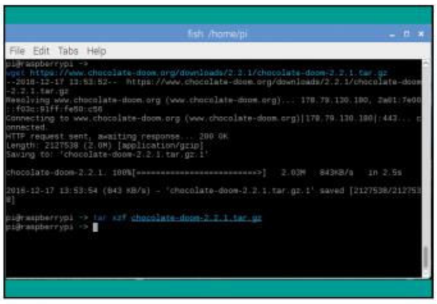
When Doom initially came out, although enthusiasts were keen to jump on the bandwagon and start creating new graphics, sound effects and so on, they couldn't do much to change some of the game's fundamental values such as how much damage was dealt by monsters. Since the release of Doom's source code and some remarkable efforts by the community at large, however, all aspects of the game can now be changed and there are thousands of WADs. Some of these are simply slight alterations to the game itself, whereas others expand hugely on the premise of the original game. One of the best known of these "MegaWADs" is Eternal Doom, which contains 32 full levels, each of which is

»

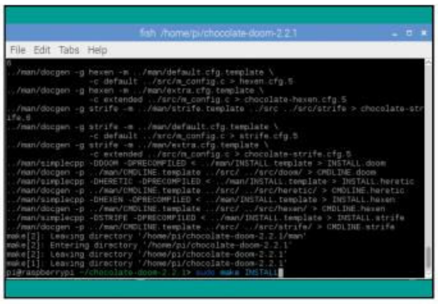
RECIPE FOR CHOCOLATE DOOM



```
pi@raspberrypi -> sudo apt-get install libsdl-mixer1.2-dev libsdl-net1.2-dev
```



```
pi@raspberrypi ->
wget https://www.chocolate-doom.org/downloads/2.2.1/chocolate-doom-2.2.1.tar.gz
--2016-12-17 13:53:52--  https://www.chocolate-doom.org/downloads/2.2.1/chocolate-doom
-2.2.1.tar.gz
Resolving www.chocolate-doom.org (www.chocolate-doom.org)... 178.79.130.180, 2a01:7e00
::f03c:91ff:fe50:c56
Connecting to www.chocolate-doom.org (www.chocolate-doom.org)|178.79.130.180|:443... c
onnected.
HTTP request sent, awaiting response... 200 OK
Length: 2127538 (2.0M) [application/gzip]
Saving to: 'chocolate-doom-2.2.1.tar.gz.1'

chocolate-doom-2.2.1. 100%[=========================>]   2.03M   843KB/s   in 2.5s

2016-12-17 13:53:54 (843 KB/s) - 'chocolate-doom-2.2.1.tar.gz.1' saved [2127538/212753
8]

pi@raspberrypi -> tar xzf chocolate-doom-2.2.1.tar.gz
pi@raspberrypi ->
```



```
../man/docgen -g hexen -m ../man/default.cfg.template \
                -c default ../src/m_config.c > hexen.cfg.5
../man/docgen -g hexen -m ../man/extra.cfg.template \
                -c extended ../src/m_config.c > chocolate-hexen.cfg.5
../man/docgen -g strife -m ../man/strife.template ../src ../src/strife > chocolate-str
ife.6
../man/docgen -g strife -m ../man/default.cfg.template \
                -c default ../src/m_config.c > strife.cfg.5
../man/docgen -g strife -m ../man/extra.cfg.template \
                -c extended ../src/m_config.c > chocolate-strife.cfg.5
../man/simplecpp -DDOOM -DPRECOMPILED < ../man/INSTALL.template > INSTALL.doom
../man/docgen -p ../man/CMDLINE.template ../src/ ../src/doom/ > CMDLINE.doom
../man/simplecpp -DHERETIC -DPRECOMPILED < ../man/INSTALL.template > INSTALL.heretic
../man/docgen -p ../man/CMDLINE.template ../src/ ../src/heretic/ > CMDLINE.heretic
../man/simplecpp -DHEXEN -DPRECOMPILED < ../man/INSTALL.template > INSTALL.hexen
../man/docgen -p ../man/CMDLINE.template ../src/ ../src/hexen/ > CMDLINE.hexen
../man/simplecpp -DSTRIFE -DPRECOMPILED < ../man/INSTALL.template > INSTALL.strife
../man/docgen -p ../man/CMDLINE.template ../src/ ../src/strife/ > CMDLINE.strife
make[2]: Leaving directory '/home/pi/chocolate-doom-2.2.1/man'
make[2]: Entering directory '/home/pi/chocolate-doom-2.2.1'
make[2]: Leaving directory '/home/pi/chocolate-doom-2.2.1'
make[1]: Leaving directory '/home/pi/chocolate-doom-2.2.1'
pi@raspberrypi ~/chocolate-doom-2.2.1> sudo make INSTALL
```

1 Install Dependencies

Your Pi will need certain programs in order to compile Chocolate Doom. Open Terminal from Applications > Accessories > Terminal or connect to the Pi via SSH and install these with the following command:

```
sudo apt-get install libsdl-mixer1.2-dev
libsdl-net1.2-dev python-imaging
```

Press Y when asked to confirm that you wish to proceed with the install.

2 Download and extract Chocolate Doom

Fire up your web browser, and head to **www.chocolate-doom.org/wiki/index.php/Downloads** to find the link for the Chocolate Doom Source Code. You can then use the `wget` command in the Pi's Terminal to download it. At the time of writing, for instance, this would be:

```
wget https://www.chocolate-doom.org/
downloads/2.2.1/chocolate-doom-
2.2.1.tar.gz
```

Once the file has downloaded, extract it using `tar` – for example:

```
tar xzf chocolate-doom-2.2.1.tar.gz
```

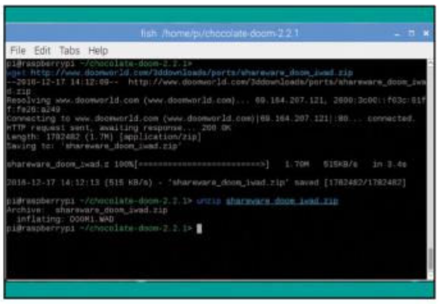
3 Compile Chocolate Doom

Use the `cd` command to switch to the newly created directory:

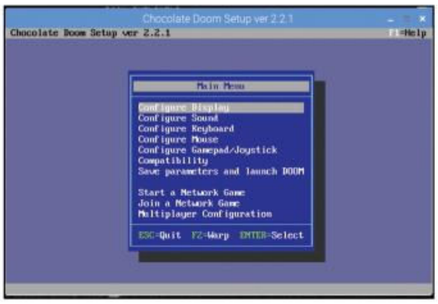
```
cd chocolate-doom-2.2.1
```

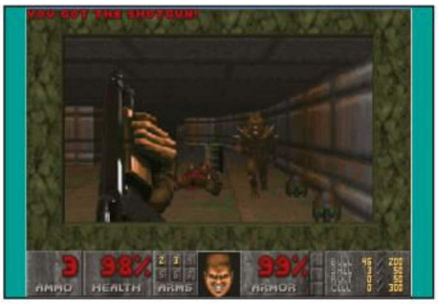
Next, run `configure` to prepare the necessary files. Now run `make` to begin compiling them. This can take some time, so feel free to take a break. Once this has completed run the following command to actually install the program:

```
sudo make install
```



```
pi@raspberrypi ~/chocolate-doom-2.2.1>
wget http://www.doomworld.com/3ddownloads/ports/shareware_doom_iwad.zip
--2016-12-17 14:12:09--  http://www.doomworld.com/3ddownloads/ports/shareware_doom_iwa
d.zip
Resolving www.doomworld.com (www.doomworld.com)... 69.164.207.121, 2600:3c00::f03c:91f
f:fe26:a249
Connecting to www.doomworld.com (www.doomworld.com)|69.164.207.121|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1782482 (1.7M) [application/zip]
Saving to: 'shareware_doom_iwad.zip'

shareware_doom_iwad.z 100%[=========================>]   1.70M   515KB/s   in 3.4s

2016-12-17 14:12:13 (515 KB/s) - 'shareware_doom_iwad.zip' saved [1782482/1782482]

pi@raspberrypi ~/chocolate-doom-2.2.1> unzip shareware_doom_iwad.zip
Archive:  shareware_doom_iwad.zip
  inflating: DOOM1.WAD
pi@raspberrypi ~/chocolate-doom-2.2.1>
```





4 Download WAD file (or files)

For the purposes of this project, we will download the shareware version of Doom, using `wget`. Make sure you are still inside the Chocolate Doom folder and run the following:

```
wget http://www.doomworld.
com/3ddownloads/ports/shareware_
doom_WAD.zip
```

Once the file has downloaded, extract it by running the following:

```
unzip shareware_doom_WAD.zip
```

You'll see that the resulting file is called DOOM1.WAD.

5 Run Doom Setup

Use the following command to launch setup:

```
chocolate-doom-setup -WAD DOOM1.
wad
```

You can use this to configure the display, sound, keyboard layout, mouse and even a gamepad if you have one.

Technically you can use this screen to host and join multiplayer Deathmatches as well, but it's actually easier just to do this from the command line. See *Doom Deathmatches* on page 50 for more information.

6 Run Chocolate Doom

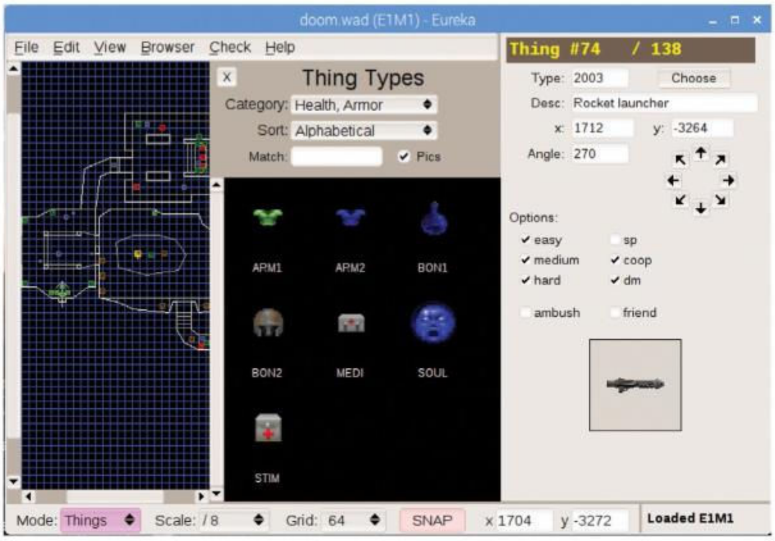
To launch Doom run the following:

```
chocolate-doom -WAD DOOM1.WAD
```

Doom will automatically launch in full-screen mode, which can place a strain on the Pi, so consider opening it in a window with this command:

```
chocolate-doom -WAD DOOM1.WAD
-window 640x480
```

If you have a different WAD file, replace DOOM1.WAD with the name of the file of your choice.



Category: Health, Armor
Sort: Alphabetical
Match:
✓ Pics
Type: 2003
Desc: Rocket launcher
x: 1712 y: -3264
Angle: 270
✓ easy ☐ sp
✓ medium ✓ coop
✓ hard ✓ dm
☐ ambush ☐ friend
Mode: Things Scale: / 8 Grid: 64 x 1704 y -3272

Click on individual items to reposition them. Use the pane on the right to change their attributes.

Pi bites

View Doomworld's entire archive of WADs at **https://www.doomworld.com/idgames**. View the Doom Community's most popular WADS at **http://www.doomworld.com/20years/best1.php**.

around four times larger than the levels in the original Vanilla Doom.

As impressive as this achievement is, Eternal is simply an expansion of the Doom Universe. Other WADs are considered "total conversions" in that they replace all resources used in the original game. The first of these was Justin Fisher's awesome Aliens TC, which is based on the eponymous film franchise.

Those interested in working in the industry should not consider designing game levels as a waste of time – Fisher was offered employment by game developer Dreamworks, which he declined in order to finish his university degree. However other designers of popular WADs such as Iikka Keränen did go on to find employment with major video game companies to work on official projects.

For copyright reasons, in this project we'll focus on the official shareware WAD of Doom (DOOM1.WAD), downloaded from fan site **Doomworld.com**. This contains only the first episode of the game, Knee Deep in the Dead, and was initially released to encourage interest in the game. Upon completion of the first level players were encouraged to order the full version.

This said, there are a number of official and unofficial WADs available for download, so provided it is legal in your jurisdiction feel free to load and experiment with these.

Freedoom isn't free

The Freedoom Project is an excellent example of the enduring popularity of Doom. The stated aim of the project is to create Doom-style WADs made entirely of free content. Since id Software have released the source code to Doom, the underlying game code is free. Freedoom builds on this with a number of levels, artwork, sound effects and music. The result is a free and open source game, devoid of any proprietary content.

Freedom provides only WAD files, so needs a compatible game engine in order to be played. The Chocolate Doom website claims that the single player Freedoom WADs are not compatible. It does however recommend downloading and playing Freedoom's set of Deathmatch levels known as "FreeDM". Unlike the regular deathmatch levels you'll find in Vanilla and Chocolate Doom, the focus is on the players – there are no monsters.

If playing Doom has whetted your appetite to try your hand at level design, the Freedoom website also has hundreds of original textures and sound effects that can be used royalty-free. Visit **https://freedoom.github.io** to find these and find out more.

Your Eureka moment

In true Raspberry Pi spirit you might be one of those who do not want to play with WADS created by others but want to edit and create levels of their own. Doom was created to be hacked, expanded and modified, so this is entirely in keeping with game's proud tradition.

Raspbian contains Eureka, the only Linux-based Doom Level Editor under development. Go to Applications > Preferences > Add/Remove Software and search for it using the search bar at the top left, or visit **https://sourceforge.net/p/eureka-editor** to download the software.

Eureka works by editing or creating WAD files, allowing you to write new levels from scratch or change the layout or objects in existing ones. You may find help with using the editor, if you need it, from the **Doomworld.com** forums.

The Doom is in the details

For the sake of simplicity, this project has focused on the shareware DOOM1.WAD, which is downloaded to the **chocolate-doom** directory. If your WAD file is located anywhere else, simply specify this when launching Chocolate Doom – for example:

```
chocolate-doom -WAD /home/pi/Downloads/
DOOM1.WAD -window 640x480
```

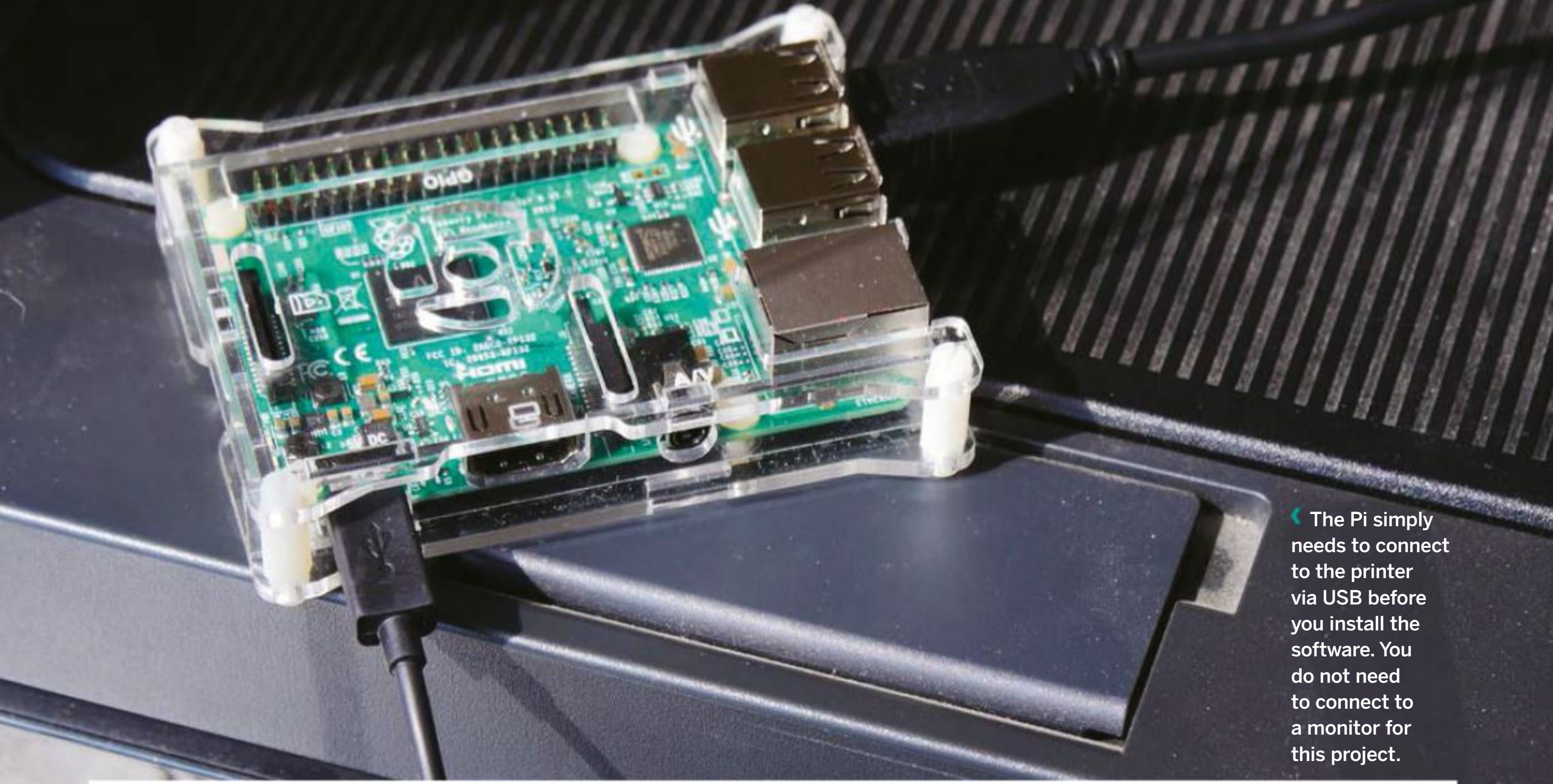
Chocolate Doom is quite pedantic about filenames (capitalisation matters), so if you are trying to locate the WAD file using the Setup screen, you may want to rename your files ".wad" rather than ".WAD". This is also true for the Eureka Level Editor (see above) – during testing the program failed to recognise the DOOM1.WAD file until it was renamed to "doom.wad".

A full list of all command line arguments and further help with Setup are available from the Chocolate Doom User Guide at **www.chocolate-doom.org/wiki/index.php/User_guide**.

In order to play deathmatches in Doom you will need to open UDP Port 2342 on your firewall and/or router. Steps to do this will vary depending on your router. Check with your manufacturer or visit the website **http://portforward.com** for steps on how to do this.

Chocolate Doom sadly is not compatible with the single player Freedoom WADS. You might find other WAD files may not work well or cause issues for your Pi. Ideally you should have a Pi dedicated to gaming so there's no risk of losing any data.

On behalf of everyone at Pi User, please also accept our warmest thanks for your efforts in fighting the forces of Hell. Good luck, soldier! ■



The Pi simply needs to connect to the printer via USB before you install the software. You do not need to connect to a monitor for this project.

Turn any printer into a wireless printer

Nate Drake explains how to transform your old (or new) USB cabled printer into a wireless printer by using a Raspberry Pi as a print server

Here's a question for you: what do Chateau Petrus 2012, petrol and milk have in common? The answer is that, measure for measure, they are all less expensive on average than printer ink.

The trend these days is that manufacturers sell reasonably priced wireless printers but rather costly ink. We do not mean to demonise printer companies – the costs of raw materials such as resin and oil have also increased in recent years – but anyone who prints at home regularly knows that this can blow a huge hole in your budget.

Some people invest in laser printers, which – while more costly up-front and typically used only for black-and-white documents – can hugely reduce your printing expenses. Older laser printers in particular are very efficient... but many do not have support for wireless, and those that do are sometimes not compatible with the most recent versions of Windows or Mac OS.

Enter the Raspberry Pi. The Pi can connect to your wireless network and also has a handy USB port, to which you can connect any printer. In this project we will explore how to connect your Pi to an old-style wired printer via USB to make it possible for anyone connected to your wireless network to print to it.

Printer preparations

One of the excellent features of this project is that it works with any model of Raspberry Pi, and you can even install additional software on the Pi after you're done. For a quick and easy setup use the Raspberry Pi 3 as this has built-in wireless.

Aside from that you will need to connect the Pi to the printer via a USB cable and switch both the Pi and the printer on before getting started.

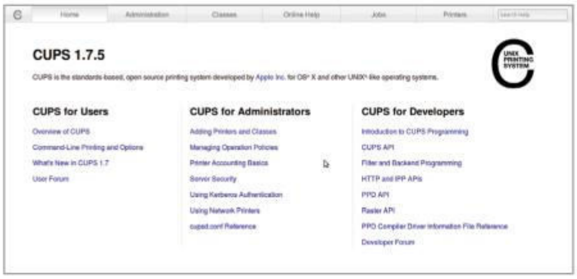
This project assumes that you are comfortable with accessing the Pi over SSH and that it is already connected to your wireless network.

Once the Pi is connected, we will install CUPS (Common Unix Printing System) in order to discover and set up your particular »

Pi bites

Remember, if you're running short of mains plug sockets and have a micro USB cable you can power your Pi from a USB socket on a computer.

The HP LaserJet Pro P1102 is relatively inexpensive but connects by USB only and is incompatible with Windows 10. Use the Pi to turn it into a working wireless printer.



CUPS 1.7.5

CUPS is the standards-based, open source printing system developed by Apple Inc. for OS® X and other UNIX®-like operating systems.

CUPS for Users
Overview of CUPS
Command-Line Printing and Options
What's New in CUPS 1.7
User Forum

CUPS for Administrators
Adding Printers and Classes
Managing Operation Policies
Printer Accounting Basics
Server Security
Using Kerberos Authentication
Using Network Printers
cupsd.conf Reference

CUPS for Developers
Introduction to CUPS Programming
CUPS API
Filter and Backend Programming
HTTP and IPP APIs
PPD API
Raster API
PPD Compiler Driver Information File Reference
Developer Forum

› CUPS is developed by Apple for its OS and other Unix based systems, such as Linux. It manages print services for some of the most popular versions such as Ubuntu Linux.

printer. You don't have to do this directly on the Pi as CUPS can be accessed over a web browser on any device connected to your home network (see *Connecting to CUPS* on the opposite page).

CUPS quite simply is a printing manager which can transform your Pi into a print server, capable of accepting print jobs across your wireless network from other devices and then sending them to your printer.

This project is focused on printers that connect over USB, but will also work for wireless printers even if they no longer work with your particular operating system. On the whole, most Linux users tend to upgrade equipment less often than software, so you may find that your computer will be able to print via CUPS even if it couldn't connect directly to your printer.

Post setup

Once you have followed the steps here and successfully printed a test page from the CUPS web interface, you will most likely want to add your new wireless printer on your computer.

The steps for doing this will vary depending on your setup. For Windows 10 users, it's best to follow the steps outlined on the Microsoft support website **http://bit.ly/Win-10-addprinter**. If your printer isn't listed, simply select the option saying so. You'll be asked to type in the exact location of your printer. You can do this by opening the CUPS web interface and clicking Manage Printers. Click on your own printer to get the right address for it, which will be something like https://192.168.1.2:631/printers/Canon_MG4200_series

If you're using a Mac, try following the steps on Apple's support website at **https://support.apple.com/en-us/HT202167**. Enter the IP address of the Raspberry Pi – for example 192.168.1.2 or whatever it is in your specific case – to set up the printer. If the Mac fails to find the printer, you may wish to follow the steps on the excellent Pi3g Blog to set up an AirPrint Service on the Pi: **http://bit.ly/Pi3g-AirPrint**.

In either case if you're asked to specify a print protocol, the CUPS manual recommends to use HP JetDirect, which is seen as the most reliable.

If you're using Linux, setup should be very simple, as CUPS was made specifically for it. However, almost every version of Linux has a different way of setting up printers. For more information, see **www.linux.com/learn/how-manage-printers-linux**.

Printer hicCUPS

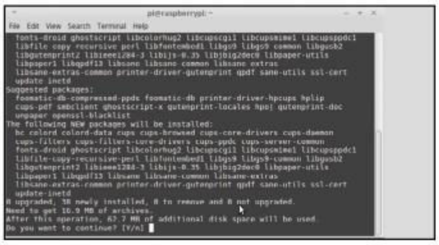
When connecting to the CUPS web interface for the first time, you might receive a warning that your connection is insecure. This is because your CUPS Admin page's security certificate has only been signed by the Pi (known as a "self-signed certificate") and not a third party certificate authority. There is no cause for concern as you will have set up the page yourself, and you can tell your web browser to make an exception for CUPS to allow you to connect in future.

Although CUPS has drivers for most printers, if it cannot find yours then you will need to locate a PPD (PostScript Printer Description File) for your

Pi bites

If you don't yet own a laser printer, try to find one that supports duplex (double-sided) printing. This saves on both energy and paper.

SET UP YOUR PI-BASED WIRELESS PRINTER



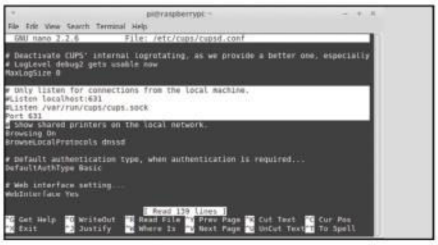
```
  fonts-droid ghostscript libcolorhug2 libcupscgi1 libcupsmime1 libcupsppdc1
  libfile-copy-recursive-perl libfontembed1 libgs9 libgs9-common libgusb2
  libgutenprint2 libieee1284-3 libijs-0.35 libjbig2dec0 libpaper-utils
  libpaper1 libqpdf13 libsane libsane-common libsane-extras
  libsane-extras-common printer-driver-gutenprint qpdf sane-utils ssl-cert
  update-inetd
Suggested packages:
  foomatic-db-compressed-ppds foomatic-db printer-driver-hpcups hplip
  cups-pdf smbclient ghostscript-x gutenprint-locales hpoj gutenprint-doc
  unpaper openssl-blacklist
The following NEW packages will be installed:
  bc colord colord-data cups cups-browsed cups-core-drivers cups-daemon
  cups-filters cups-filters-core-drivers cups-ppdc cups-server-common
  fonts-droid ghostscript libcolorhug2 libcupscgi1 libcupsmime1 libcupsppdc1
  libfile-copy-recursive-perl libfontembed1 libgs9 libgs9-common libgusb2
  libgutenprint2 libieee1284-3 libijs-0.35 libjbig2dec0 libpaper-utils
  libpaper1 libqpdf13 libsane libsane-common libsane-extras
  libsane-extras-common printer-driver-gutenprint qpdf sane-utils ssl-cert
  update-inetd
0 upgraded, 38 newly installed, 0 to remove and 0 not upgraded.
Need to get 16.9 MB of archives.
After this operation, 62.7 MB of additional disk space will be used.
Do you want to continue? [Y/n]
```

1 Install CUPS and other useful packages

Open Terminal on your Pi or connect via SSH and run the command:

```
sudo apt-get install avahi-daemon
cups cups-pdf python-cups
```

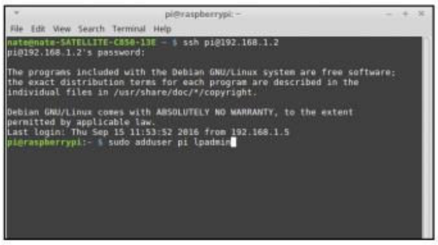
The installation process will take some time as CUPS downloads drivers for some common printers.



```
  GNU nano 2.2.6          File: /etc/cups/cupsd.conf

# Deactivate CUPS' internal logrotating, as we provide a better one, especially
# LogLevel debug2 gets usable now
MaxLogSize 0

# Only listen for connections from the local machine.
#Listen localhost:631
#Listen /var/run/cups/cups.sock
Port 631
# Show shared printers on the local network.
Browsing On
BrowseLocalProtocols dnssd

# Default authentication type, when authentication is required...
DefaultAuthType Basic

# Web interface setting...
WebInterface Yes

                         [ Read 139 lines ]
^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text^T To Spell
```

2 Modify CUPS to allow Remote Access

This step is optional but recommended as it allows you to access the CUPS web interface from another browser on your network. See "Connecting to CUPS" for information.



```
nate@nate-SATELLITE-C850-13E ~ $ ssh pi@192.168.1.2
pi@192.168.1.2's password:

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Sep 15 11:53:52 2016 from 192.168.1.5
pi@raspberrypi:~ $ sudo adduser pi lpadmin
```

3 Add your user to admin group

Again in Terminal or over SSH enter the following command, replacing "yourusername" with the appropriate user name:

```
sudo adduser yourusername lpadmin
```

This will make sure you can administer printers.

CONNECTING TO CUPS

By default, when you install CUPS on your Raspberry Pi, the web interface can be accessed only on the Pi itself. This can be a nuisance if you cannot connect the Pi to the printer, keyboard/mouse and a TV screen at the same time.

Fortunately it is possible to modify the CUPS configuration file to allow access from other devices on your home network. First connect to your Pi over SSH and run the following command:

```
sudo nano /etc/cups/cupsd.conf
```

Scroll down until you find the text "Only listen for connections from the local machine". Type a **#** at the start of each of the next two lines to comment them out, then on a new line after this, type the following text (see the illustration):

```
Port 631
```

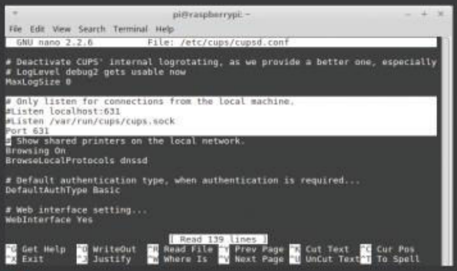
Next, scroll down to the section reading "Restrict Access to the Server" and look for the words "Order allow, deny". Put a new line after this with the following words:

```
Allow @Local
```

Scroll down to the next section headed "Restrict Access to the Admin Pages" and once again after the text "Order allow, deny" place the words "**Allow @Local**". Finally scroll down to the next section, reading "Restrict Access to Configuration files", and again insert "**Allow @Local**" after the text "Order allow, deny".

Press Ctrl+X, then Y, then Return to save and exit. Restart the CUPS service to apply your changes with the following command:

```
sudo service cups restart
```


pi@raspberrypi: ~
File Edit View Search Terminal Help

```
  GNU nano 2.2.6              File: /etc/cups/cupsd.conf

# Deactivate CUPS' internal logrotating, as we provide a better one, especially
# LogLevel debug2 gets usable now
MaxLogSize 0

# Only listen for connections from the local machine.
#Listen localhost:631
#Listen /var/run/cups/cups.sock
Port 631
# Show shared printers on the local network.
Browsing On
BrowseLocalProtocols dnssd

# Default authentication type, when authentication is required...
DefaultAuthType Basic

# Web interface setting...
WebInterface Yes

                          [ Read 139 lines ]
^G Get Help   ^O WriteOut   ^R Read File  ^Y Prev Page  ^K Cut Text   ^C Cur Pos
^X Exit       ^J Justify    ^W Where Is   ^V Next Page  ^U UnCut Text ^T To Spell
```

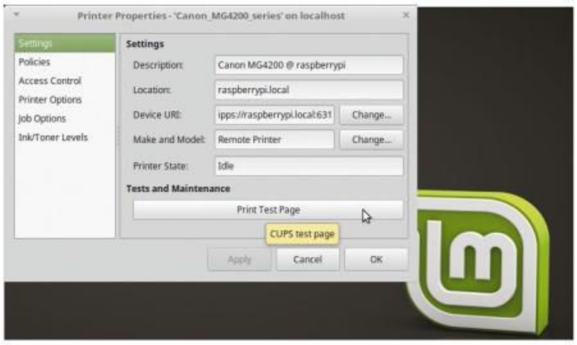
To enable remote access to CUPS, first type a hash sign on the two lines after "Only listen for connections from the local machine".

You will now be able to access the CUPS interface from any web browser connected to your local network by visiting **https://yourpiIPaddress:631**.

printer. This file simply details your printer's specifications such as paper size, available fonts, resolutions and so on. You may be able to download this file from the printer manufacturer's website or get help by contacting its technical support number.

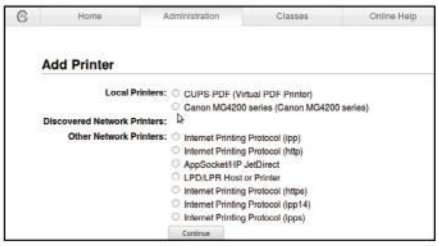
If no driver is available, or you still haven't decided which printer to buy, the Open Printing website, which is affiliated with the Linux Foundation, has an extensive database of printers that are compatible with Linux and therefore should be compatible with CUPS running on the Pi. Visit **www.openprinting.org/printers** for more information.

If your device is a multifunction device printer-scanner or you also have a USB scanner you'd like to use wirelessly, then you may also be able to scan wirelessly by downloading and installing SANE (Scanner Access Now Easy). The Pi3g Blog has an excellent tutorial on this at **http://bit.ly/Pi3g-scanner-share**. ■


Printer Properties - 'Canon_MG4200_series' on localhost
Settings
Policies
Access Control
Printer Options
Job Options
Ink/Toner Levels
Settings
Tests and Maintenance
Print Test Page
CUPS test page
Change...
Change...
Apply
Cancel
OK

Description: Canon MG4200 @ raspberrypi
Location: raspberrypi.local
Device URI: ipps://raspberrypi.local:631
Make and Model: Remote Printer
Printer State: Idle

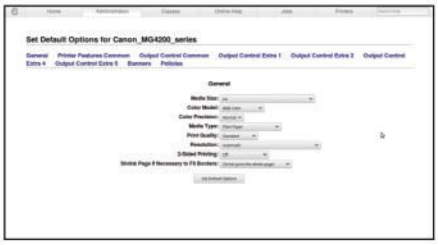
After you've got everything set up, you'll be able to manage your printer just as if it were connected directly to your computer.


Home
Administration
Classes
Online Help
Add Printer
Continue

Local Printers: ○ CUPS PDF (Virtual PDF Printer)
○ Canon MG4200 series (Canon MG4200 series)
Discovered Network Printers:
Other Network Printers: ○ Internet Printing Protocol (ipp)
○ Internet Printing Protocol (http)
○ AppSocket/HP JetDirect
○ LPD/LPR Host or Printer
○ Internet Printing Protocol (https)
○ Internet Printing Protocol (ipp14)
○ Internet Printing Protocol (ipps)

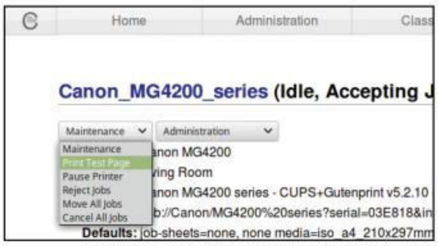
4 Connect to CUPS interface and add printer

In a browser go to the CUPS interface by visiting the address **https://yourpiIPaddress:631** (for example **https://192.168.1.2:631**). Click the grey Add Printer button on the left-hand side of the screen. By default the username is "pi" and the password "raspberry". With luck you'll see your printer. Select it, then click Continue.


Set Default Options for Canon_MG4200_series
General
Printer Features Common
Output Control Common
Output Control Extra 1
Output Control Extra 2
Output Control Extra 4
Output Control Extra 5
Banners
Policies
General
Media Size:
Color Model:
Color Precision:
Media Type:
Print Quality:
Resolution:
2-Sided Printing:
Shrink Page If Necessary to Fit Borders:

5 Configure Default Options

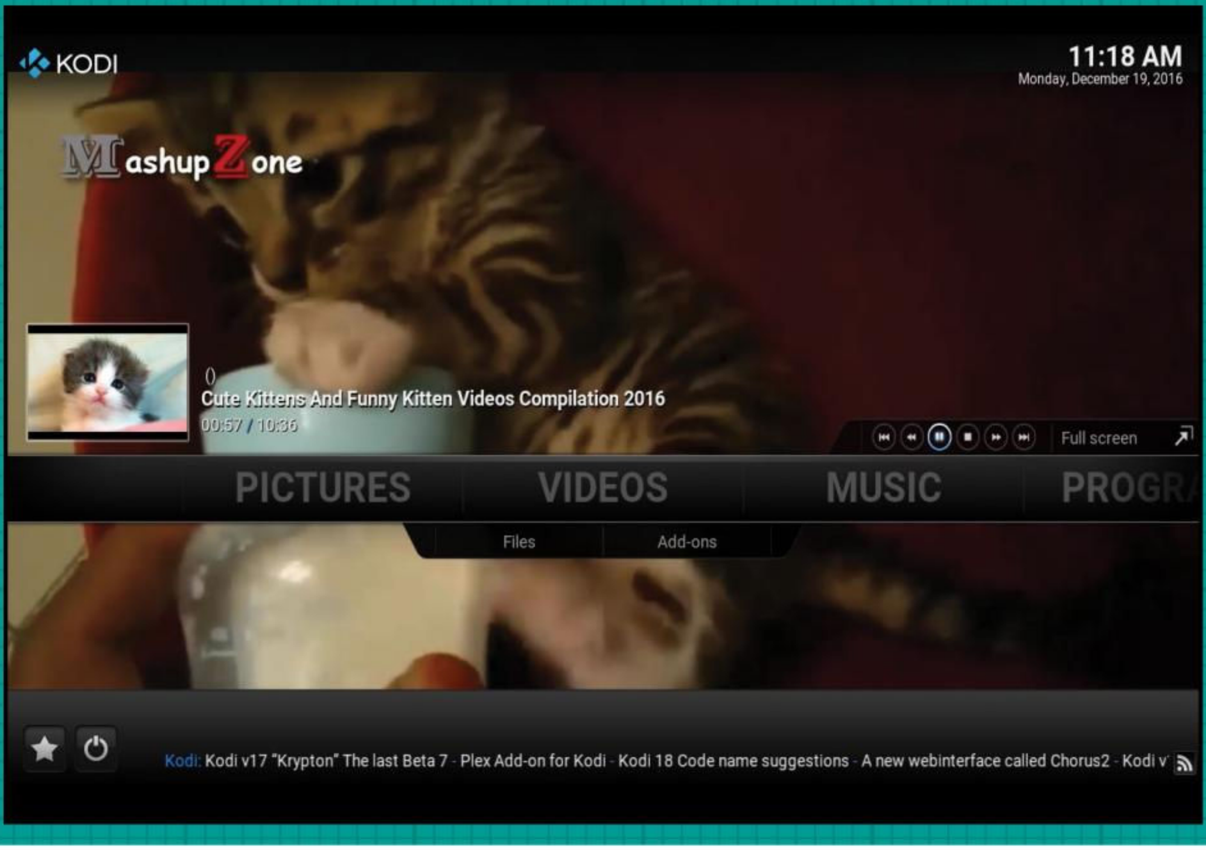
Provided you selected the correct driver in the previous step, the default options such as paper size, ink type and so on should be correct, but feel free to change these if need be. Once you're happy, click Set Default Options to complete setup.


Home
Administration
Class
Canon_MG4200_series (Idle, Accepting J
Maintenance
Administration
Maintenance
Print Test Page
Pause Printer
Reject Jobs
Move All Jobs
Cancel All Jobs

anon MG4200
ving Room
anon MG4200 series - CUPS+Gutenprint v5.2.10
b://Canon/MG4200%20series?serial=03E818&in
Defaults: job-sheets=none, none media=iso_a4_210x297mm

6 Print test page

This step is optional but an excellent idea before you try to add the printer to your home network. To verify that everything is working, click the Maintenance menu at the top-left and choose Print Test Page.


KODI
11:18 AM
Monday, December 19, 2016
Mashup Zone
Cute Kittens And Funny Kitten Videos Compilation 2016
00:57 / 10:36
Full screen
PICTURES
VIDEOS
MUSIC
PROGRA
Files
Add-ons
Kodi: Kodi v17 "Krypton" The last Beta 7 - Plex Add-on for Kodi - Kodi 18 Code name suggestions - A new webinterface called Chorus2 - Kodi v

› A preview of things to come. Kodi's menus are very easy to navigate. Videos will continue to play in the background if you've already selected them.

Create a Pi-based Media Centre

We show you how to make your Pi a one-stop media centre – for movies, TV shows, music and internet streams – through the power of LibreELEC + Kodi

Wouldn't it be great if you could have a smart home media centre for a fraction of the cost of an off-the-shelf system? Thanks to LibreELEC and the Raspberry Pi, you can now have your own fully-featured media centre for your TV shows, films and music, with support for internet streaming via YouTube and other services.

LibreELEC describes itself as "just enough OS for Kodi". It's a free, fast, open source Operating System which is available to install on any SD card using the handy OS installation manager NOOBS. It is a complete media centre suite which comes with a modified version of Kodi (formerly known as XBMC), an open source media player capable of streaming media from the internet as well as playing downloaded files.

As confusing as this terminology may be, the actual setup process is very simple. You can have a fully-featured media centre set up in minutes.

For this project, you'll need a microSD card with NOOBS installed. You can make this yourself by following the instructions on the Raspberry Pi website (**www.raspberrypi.org/documentation/installation/noobs.md**) or buy an SD card with NOOBS pre-installed for around £6 from Pi dealers and websites such as **www.thepihut.com**.

We also strongly recommend that you use a Raspberry Pi 3, as it has the best support for audio and video playback. You can use your existing keyboard if you wish but true couch potatoes may prefer to opt to use a remote (see *Taking Control* on the opposite page).

You will also need to connect your Pi via HDMI to a television for this project, both for setup with NOOBS and for media playback. You can connect to the Pi later via SSH to change the configurations or add codecs remotely if you wish (see *Kodi Codecs* over the page).

Kodi conventions

Where possible Kodi will try to display information about movies and TV shows in the database such as a plot summary, episode name, trailer and so on. It does this through use of built-in "scrapers" – small programs that analyse videos and gather the above information from online movie and TV show databases such as IMDB and TheTVDB. In order to help Kodi display the correct details, you should name movie files both with the correct title and year, for instance:

```
\Movies\Night of the Living Dead (1968).avi
```

The year is especially important if you wish to distinguish between remakes of an original film. Night of the Living Dead, for instance, was remade in 1990.

Use folders to organise your collection of TV shows. Individual episodes follow the format "<name> seasonNNepisodeNN" – for example:

```
\TV Shows\Adventures of Superman\superman_s01e03
```

You can find more detailed information on naming conventions in the Kodi Wiki at **http://kodi.wiki/view/Naming_video_files**.

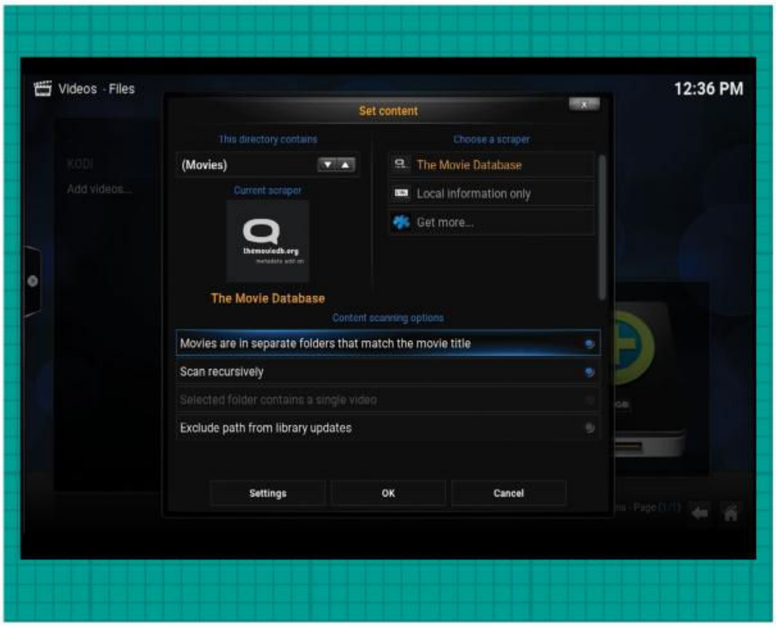
Adding your media files

Once you've named your downloaded files, copy them to an external drive so that they can be accessed by your Raspberry Pi. Provided you have abided by the naming conventions above, it doesn't really matter how you organise your collection – for the purposes of this project we used an external hard drive named "KODI", inside which there are three folders for "Movies", "TV Shows" and "Music".

The initial setup process is easiest using the keyboard and mouse. You can add more files later using the remote.

To add movies. First click on Videos, then Files in the main menu. Choose Add Videos from the menu on the left, then click Browse in the window that appears at the top right. You'll now need to navigate to your external drive. On the Pi this is usually found by clicking Root FileSystem, then Media and searching in the Pi folder.

Select the Movies folder, then click OK. The Add Video Source window will appear. At this stage you can type a new name for the media source, or just stick with Movies. Next click OK. The Set Content window will open. Click on the drop-down menu below the words "This directory contains" at the top left. Click on the small arrows to select Movies.



This directory contains: (Movies)

Choose a scraper: The Movie Database; Local information only

The Movie Database

Content scanning options

Movies are in separate folders that match the movie title

Scan recursively

Selected folder contains a single video

Exclude path from library updates

Set "This directory contains" to the correct media type, in this case Movies.

If you've organised your movies into separate folders, click the option reading "Movies are in separate folders that match the movie title" and click OK. Otherwise simply click OK to continue.

You'll be taken back to the Files section of the Video menu. Click Movies to see your films. Right-click on any film and select See Information to view the film poster, read a plot summary and even play the trailer if available via YouTube.

Adding TV shows works in exactly the same way except that in the Set Content window you should set the "This directory contains" menu to TV Shows. Do this by clicking the white up arrow at the right of the menu.

»

Pi bites

You can return to the Set Content window later to change the settings for individual folders. Visit **http://kodi.wiki/view/Set_content_and_scan** for more information.

TAKING CONTROL OF KODI

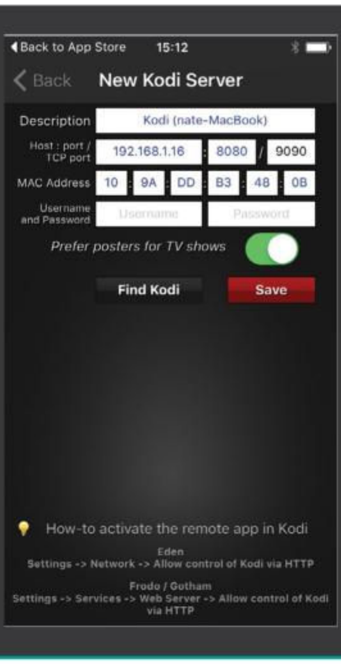
Although you can keep using your Pi's existing keyboard to control Kodi,this isn't particularly elegant, especially considering that the software is designed to be run with a simple remote control.

If you're willing to part with £20, the Pi Hut sells the handy Flirc USB device. In simplest terms it is a Universal Infra-Red Receiver. It connects to your Pi and in combination with a smart application can "learn" infra-red commands from your existing TV remote, allowing you to control both the television and the Pi. If you don't have a remote, the Pi Hut also sells a small one which can be used with the Flirc USB device.

If the tactile sensation of a remote (or maybe the clutter of another device lying around) isn't for you, Android and iPhone users will be pleased to hear there's an official Kodi Remote app in their respective online stores.

Setting up is similar no matter what device you use. After installation of Official Kodi Remote, launch it from your device's home screen. Click Add Host at the bottom left of the screen to go to the Kodi New Server screen. If you have not done so already, enable remote access to your Kodi server as outlined under "Remote Access" over the page.

Enter anything you wish in the "Description" field at the top. Next type the IP address and the port number (default is 8080) of your Pi. Click the Find Kodi button to have the app automatically find your Pi. You will now be able to search, scan and play content via your device.



Description: Kodi (nate-MacBook)

Host : port / TCP port: 192.168.1.16 : 8080 / 9090

MAC Address: 10 : 9A : DD : B3 : 48 : 0B

Username and Password

Prefer posters for TV shows ●

How-to activate the remote app in Kodi

Eden
Settings -> Network -> Allow control of Kodi via HTTP

Frodo / Gotham
Settings -> Services -> Web Server -> Allow control of Kodi via HTTP

❯ Enter your Pi's IP address (find this with the `ifconfig` command) and Port 8080, then click Find Kodi to automatically search for your Pi. Be sure to enable remote access first from the Pi.

KODI CODECS

The licences to use the AVI and WMV codecs can be bought inexpensively from the Raspberry Pi website. The MPEG-2 codec, which allows playback of .avi files, costs £2.40, and the VC-1 codec, which allows playback of .wmv files, is £1.20. Bear in mind that there are a number of media files LibreELEC will play out-of-the-box and you will not need these codecs to stream video through YouTube and so on.

In order to proceed you will need your Pi's serial number. To get this, open Terminal on the Pi or connect via SSH and run the following command:

```
cat /proc/cpuinfo
```

Copy down the 16 digit number next to "Serial".

Next visit **www.raspberrypi.com/license-keys** and click on the name of the codec you'd like to buy. Enter your serial number in the box marked "Serial", then Add to Cart. Your licence key for the codec will be sent to you via e-mail within 72 hours.

Once you have received the licence key, connect to your Pi via SSH once again. Use the following commands to open your configuration file:

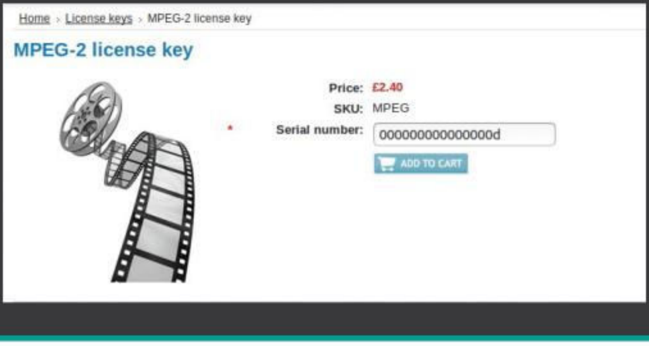
```
cd /boot
sudo nano config.txt
```

Next paste your licence code(s) at the bottom in the following format:

```
decode_MPG2=0x00000000
decode_WVC1=0x00000000
```

Replace the "0x00000000" in each case with the actual licence key you received by e-mail.

Press Ctrl + X, Y, then Return to save and exit. Reboot your Pi for the changes to take effect. Kodi will now be able to play back files in these formats.



Price: £2.40
SKU: MPEG
Serial number: 000000000000000d

Paste your Pi's serial number, then click Add to Cart for each codec you want to purchase.

Pi bites

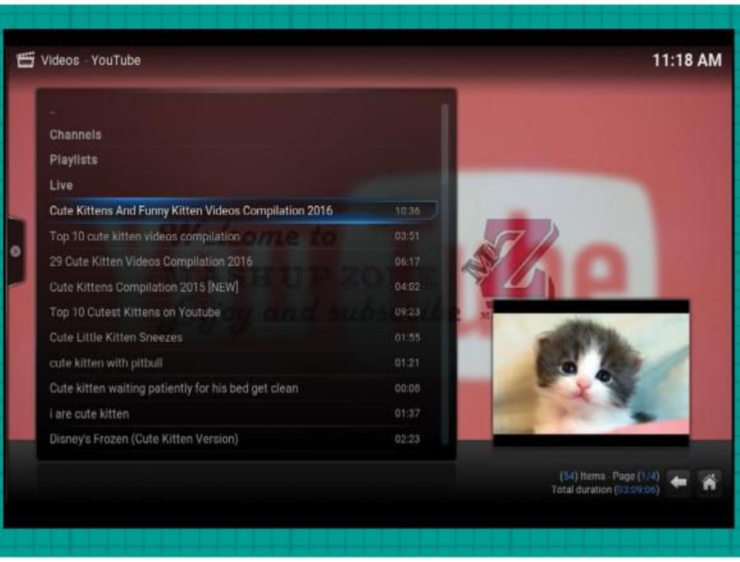
The LibreELEC website also has a handy tool to automatically download the OS to an SD Card. See **http://bit.ly/LibreELEC-SD** for more information

Kodi will automatically check for new content and use scrapers to download the necessary metadata for you each time it starts.

A word on add-ons

Kodi can download enhancements to allow it do much more than play media files. These can be installed via the AddOn Manager. For instance to install the YouTube add-on, go to the Kodi main menu and go to System > Settings. Next choose Add Ons from the menu on the left. Scroll down to Video Add Ons, select this, then go through the list to select YouTube, then click Install.

Once the installation is complete you can access YouTube from the main menu by going to Videos > Add Ons, then clicking YouTube. The first time YouTube runs within Kodi, you will be asked to sign in and choose your language and region. From there you can select a video in your playlist or search for others.



Cute Kittens And Funny Kitten Videos Compilation 2016 10:36
Top 10 cute kitten videos compilation 03:51
29 Cute Kitten Videos Compilation 2016 06:17
Cute Kittens Compilation 2015 [NEW] 04:02
Top 10 Cutest Kittens on Youtube 09:23
Cute Little Kitten Sneezes 01:55
cute kitten with pitbull 01:21
Cute kitten waiting patiently for his bed get clean 00:08
i are cute kitten 01:37
Disney's Frozen (Cute Kitten Version) 02:23

The YouTube search in action. Complete the setup wizard to access the main menu and look for fluffy kittens.

The YouTube add-on is one among hundreds available. Browse the "Look and Feel" category to select a different skin for Kodi. You can also add automatic downloads for subtitles from popular websites such as OpenSubtitles.org from the Subtitles category.

If you do choose to install add-ons, try to make sure you install just one for each purpose. For instance there's no need to have more than one scraper for movies and it can cause errors.

Remote Access

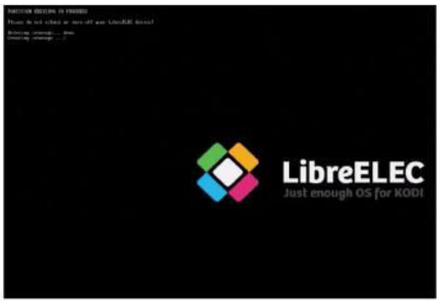
The Kodi server can be accessed remotely via SSH and Samba file sharing, but this is more for adding new content and maintenance. If you wish to access the server in order to play content, head to System > Settings > Services > Web Server. Click the option to "Allow Remote Control via HTTP". You may set a password here if you wish but by default it's blank. The web interface is quite simplistic by default but can be used to have your Pi play content without using a keyboard or mouse.

If you want to use the official Kodi Remote App on your mobile device, you should also click the Remote Control tab in this Settings > Services window and highlight the option to "Allow Remote Control by programs on other systems". In order to set this up, you will need to know your Pi's IP address. To view this, connect to the Pi via SSH and run the command `ifconfig`. Alternatively install the handy app Fing to your mobile device, which will list the IP addresses of all devices connected to your network.

AirPlay support

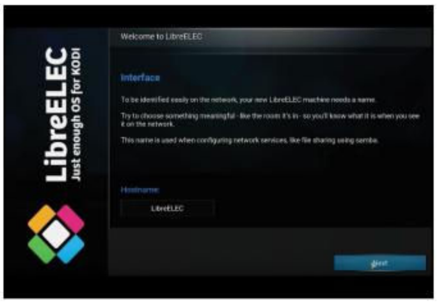
Thanks to changes made by Apple to its wireless protocol, AirPlay is not very well supported any more in Kodi. Users of iOS 9 or newer devices can stream downloaded music via AirPlay by going to

SETTING UP YOUR MEDIA CENTRE


LibreELEC
Just enough OS for KODI

1 Install LibreELEC

Connect your Pi to a monitor and restart holding down the Shift key. The NOOBS menu will provide you a choice of OS to install. Tick LibreELEC. If you don't see this option, make sure your Pi is connected to the internet by clicking Wireless Networks. Confirm that you understand installing LibreELEC will delete anything that's already on the SD card.


LibreELEC
Just enough OS for KODI
Welcome to LibreELEC
Interface
Hostname
Next

To be identified easily on the network, your new LibreELEC machine needs a name.

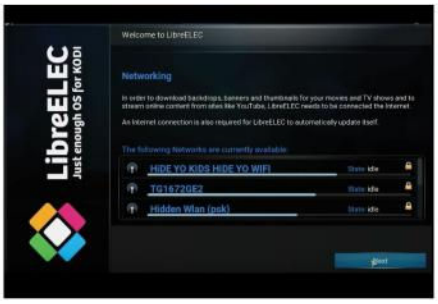
Try to choose something meaningful - like the room it's in - so you'll know what it is when you see it on the network.

This name is used when configuring network services, like file sharing using samba.

Hostname: LibreELEC

2 Set Device Name

Once the installation is complete, the LibreELEC setup wizard will appear. Click Next to continue to the first step, which is to set a meaningful name for your device. If you plan on having more than one media centre, consider numbering them, along the lines of "jane-librelec-1". This will also help identify the device on the network if you use file sharing later.

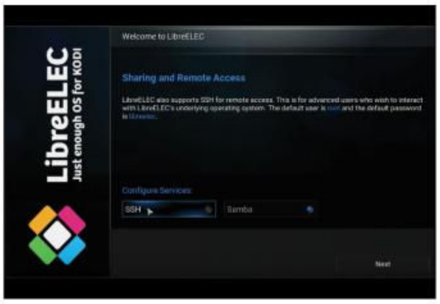

LibreELEC
Just enough OS for KODI
Welcome to LibreELEC
Networking
The following Networks are currently available
HIDE YO KIDS HIDE YO WIFI
TG1672GE2
Hidden Wlan (psk)
State idle
State idle
State idle
Next

In order to download backdrops, banners and thumbnails for your movies and TV shows and to stream online content from sites like YouTube, LibreELEC needs to be connected the Internet.

An Internet connection is also required for LibreELEC to automatically update itself.

3 Set up networking

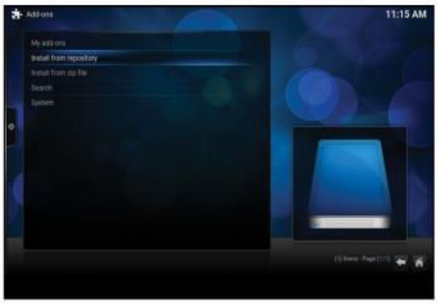
If your Pi is connected to your router via Ethernet cable, this window will not appear, so you can ignore this step. Otherwise LibreELEC will inform you that it needs internet access to download data for movies and TV shows as well as to run updates. Choose your network from the available wireless networks and enter your Wi-Fi password.


LibreELEC
Just enough OS for KODI
Welcome to LibreELEC
Sharing and Remote Access
Configure Services
SSH
Samba
Next

LibreELEC also supports SSH for remote access. This is for advanced users who wish to interact with LibreELEC's underlying operating system. The default user is root and the default password is libreelec.

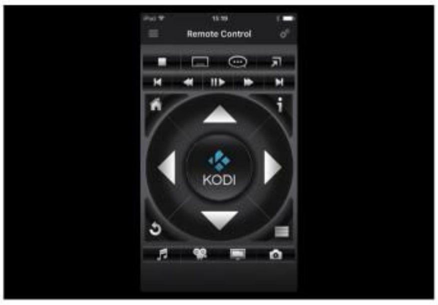
4 Set up sharing

This step is optional but can be helpful if you want to access your Pi via another computer. Enable the ability to connect via SSH and/or Samba by highlighting the radio button next to each. If you choose to connect via SSH the default username is "root" and the password is "libreelec". You can change these to something more secure if you wish.


Add-ons
11:15 AM
My add-ons
Install from repository
Install from zip file
Search
System

5 Install Add-Ons

This step is also optional but recommended if you want to add free streaming TV channels or websites to Kodi. Official add-ons can be accessed from System > Settings > Add Ons. Follow the steps outlined under "A word on add-ons" (on the opposite page) to enable support for YouTube. Install extra codecs now if necessary (see *Kodi codecs* opposite).


Remote Control
KODI

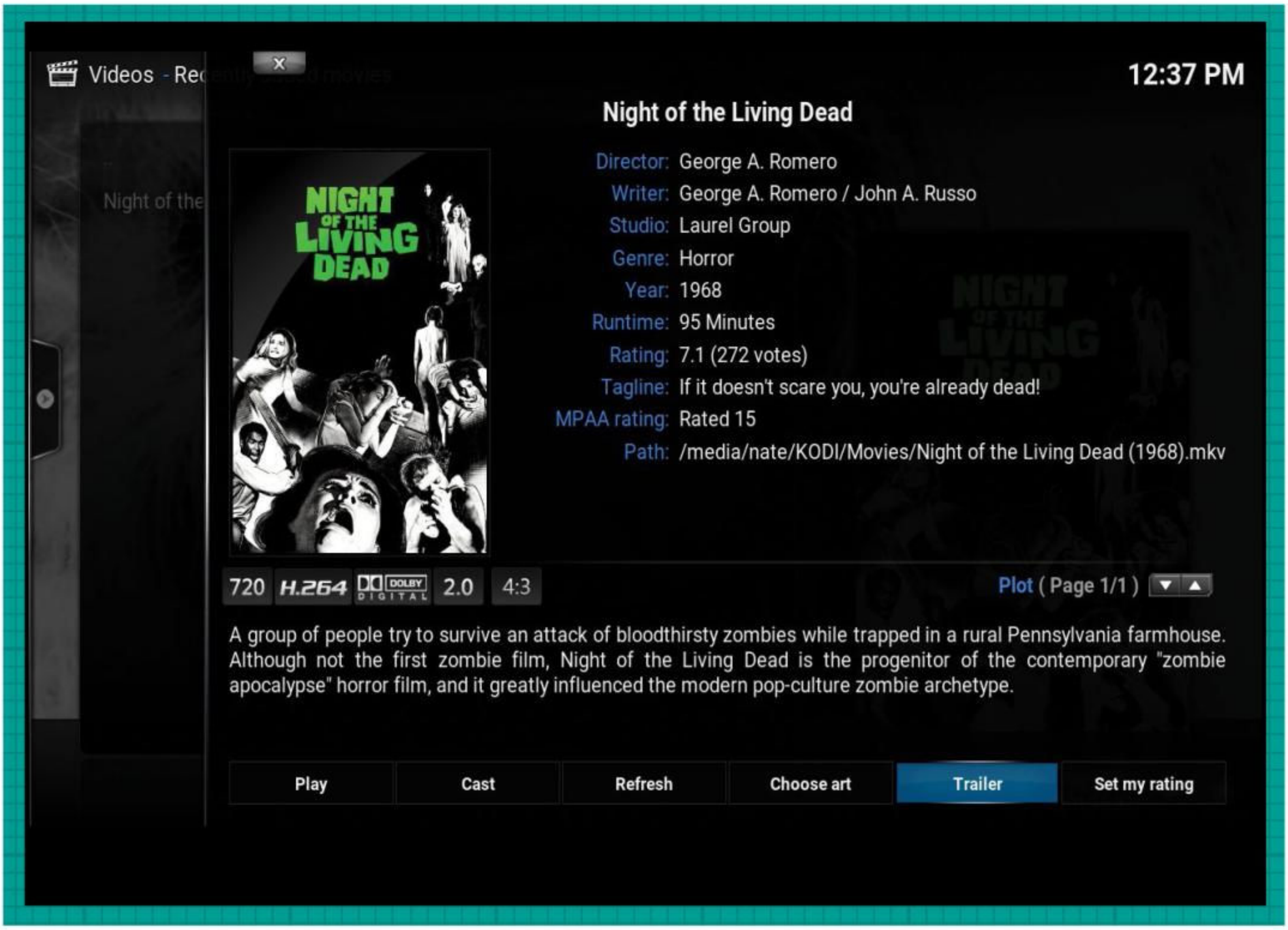
6 Set up remote control

If you have a Flirc USB device, you need to set it up using a different computer. Download software for Windows, Mac or Linux from **https://flirc.tv/downloads**. Next go to **https://support.flirc.tv** and click "How to Set up Flirc" for instructions on how to calibrate your remote. If you prefer, download the Kodi remote app to your mobile device. Further help is available from **http://kodi.wiki/view/Official_Kodi_Remote**.

System > Settings > Services in Kodi and changing the Settings Level at the bottom left. Click the AirPlay tab on the left. Click "Enable AirPlay Support" and disable "Enable AirPlay 'Videos' and 'Pictures' Support". The Pi should now appear as an AirPlay device on your Apple devices. AirPlay support is patchy, though, depending on your version of iOS, so do not be surprised if this doesn't work. Visit **http://kodi.wiki/view/AirPlay** for more information.

Be warned – installing LibreELEC will replace the Raspbian OS on your microSD card, along with any files you created and/or downloaded. Either obtain a fresh SD card with NOOBS installed or back up any files on your existing card before setting up your media centre.

Should Kodi be unable to see your external drive, this may be because the drive requires more voltage than the Pi's USB ports can deliver. Try again using a powered USB hub. »


Videos - Recently added movies
12:37 PM
Night of the Living Dead
NIGHT OF THE LIVING DEAD
720 H.264 DOLBY DIGITAL 2.0 4:3
Plot (Page 1/1)
Play
Cast
Refresh
Choose art
Trailer
Set my rating

Night of the Living Dead

Director: George A. Romero
Writer: George A. Romero / John A. Russo
Studio: Laurel Group
Genre: Horror
Year: 1968
Runtime: 95 Minutes
Rating: 7.1 (272 votes)
Tagline: If it doesn't scare you, you're already dead!
MPAA rating: Rated 15
Path: /media/nate/KODI/Movies/Night of the Living Dead (1968).mkv

A group of people try to survive an attack of bloodthirsty zombies while trapped in a rural Pennsylvania farmhouse. Although not the first zombie film, Night of the Living Dead is the progenitor of the contemporary "zombie apocalypse" horror film, and it greatly influenced the modern pop-culture zombie archetype.

Pay close attention to file names to make sure the metadata is correct. Place the year of the film in brackets after the title. The year 1968 has been added to the above film to distinguish it from the 1990 remake.

If you have issues with playing certain video files, this may be due to missing codecs. These are computer programs for encoding/decoding large data files such as video. There are hundreds of different codecs but they're generally considered a necessary evil as they help to compress media files to a size where they can be easily managed and shared over the internet.

Codec confusion

Some codecs are free, open and unpatented, such as Ogg Vorbis, which can be used to encode music files. As a result, computer manufacturers can bundle them with devices so they can be played out of the box. Other codecs are proprietary, in that they can be downloaded and used only on payment of a licence fee. In order to support the principles of free and open software (and keep costs down), the Raspberry Pi doesn't include such codecs.

In practice this can mean that if you try to play media files in quite common formats such as Windows Media Video (.wmv) or AVI (.avi), you may find LibreELEC doesn't have the right codec to play them. Follow the steps in *Kodi Codecs* on page 58 to download and install them.

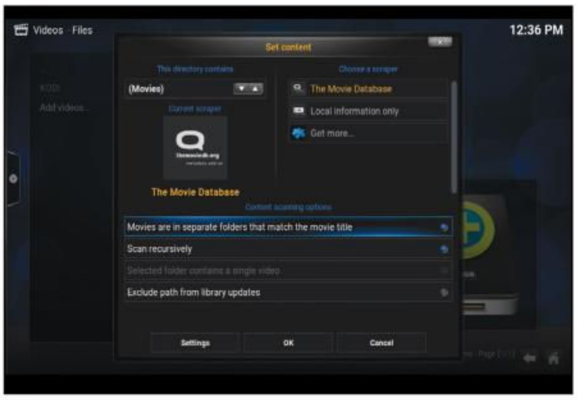
If movies and TV shows you add do not display correctly, make sure you have followed the naming conventions as laid out in the Kodi Wiki, being sure to include the movie year.

If you're having video or audio issues, from the main Kodi menu go to System > Settings and select either "Video Output" or "Audio Output" to adjust your resolution or change your sound channels, among other options.

Finally, make sure that you either have the permission of the copyright holder to download any films or TV shows, or that they are in the Public Domain. Because of differing copyright laws there is no one definitive list of public domain films, but the Internet Archive (**www.archive.org**) is an excellent place to start. ■

Pi bites

Select the Programs category, then "Get More" from the Kodi main menu for extra apps to check your e-mail, post to Facebook, chat on IRC and much more.


Videos - Files
12:36 PM
Set content
This directory contains
Choose a scraper
KODI
Add videos...
Current scraper
themoviedb.org
The Movie Database
Local information only
Get more...
Content scanning options
Settings
OK
Cancel

(Movies)

The Movie Database

Movies are in separate folders that match the movie title
Scan recursively
Selected folder contains a single video
Exclude path from library updates

Kodi can automatically check for new content and even use scrapers, mini-programs which download metadata for TV shows and films for you. The default scraper is The Movie Database (themoviedb.org).

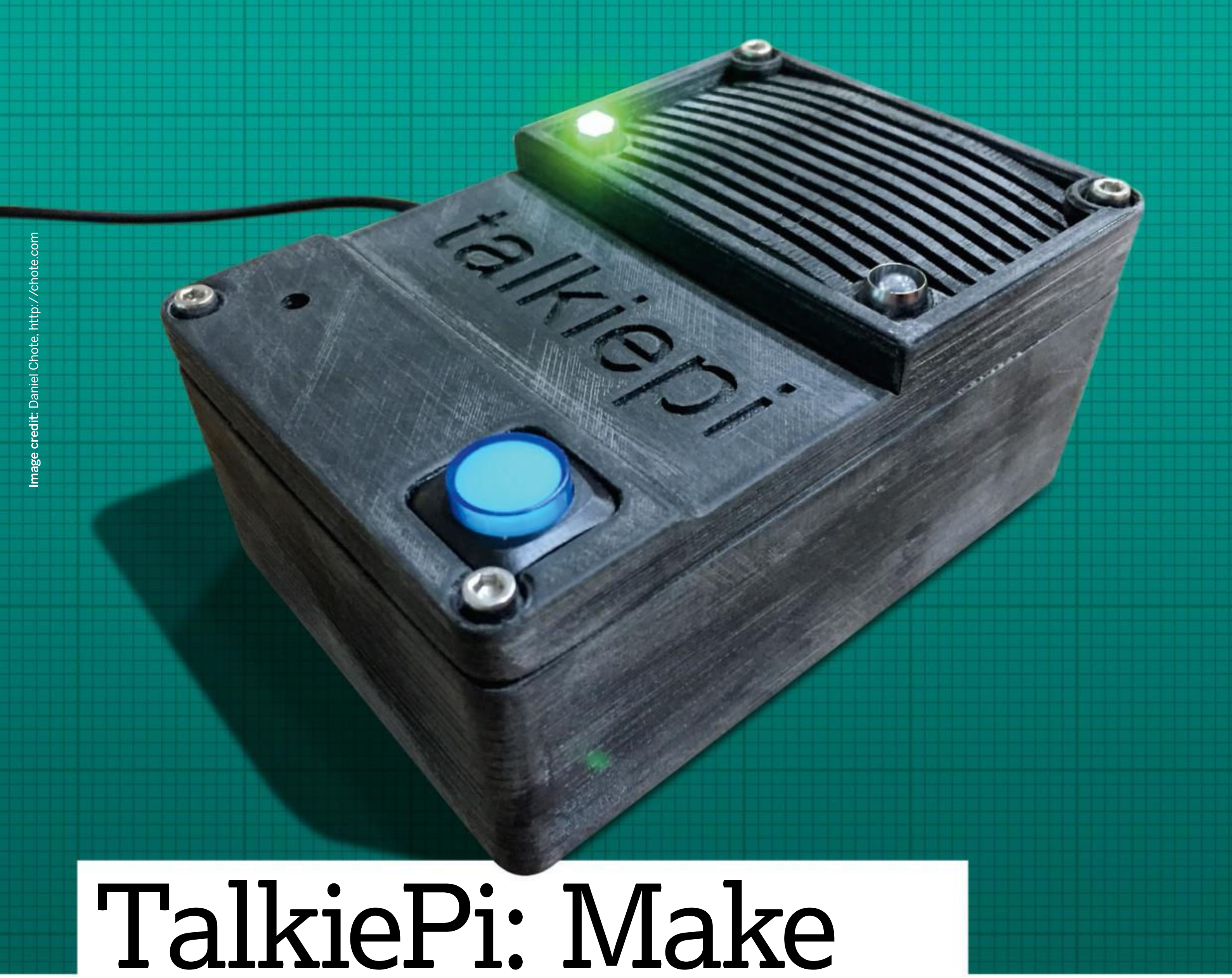


Image credit: Daniel Chote, http://chote.com

TalkiePi: Make a walkie-talkie

Nate Drake tickles your nostalgia bone with Daniel Chote's awesome TalkiePi project to create a retro-style Wi-Fi walkie-talkie

Those who have been following the awesome TV series Stranger Things may have felt a twinge of nostalgia at the retro walkie-talkies used by the children. Fortunately, New Zealander Daniel Chote decided to take his own homage to the '80s one step further. The TalkiePi, which is his brainchild, is a project to turn your Pi into a simple push-button walkie-talkie which works over Wi-Fi.

Chote designed the project with his kids in mind, hence the simple interface. By default, once the software is installed it will automatically connect to his own server using the Mumble protocol, enabling users to chat right away. Along with the software on his GitHub page are detailed instructions about components and even a casing design that can be produced with a 3D printer to provide a nostalgic experience that will tug at the heartstrings of the average 30-something or older.

The interface of the TalkiePi couldn't be simpler. When the button is depressed, it lights up and you can talk. When it's released the channel is clear. There are two other status LEDs, indicating your connection to the wireless network and whether others have joined your particular channel.

Clearly, this project is a labour of love but it isn't useful only to children. Many gamers make use of the Mumble protocol to talk privately while playing. Or two TalkiePi's can also be used as a handy intercom for your front door.

Getting started

To begin you'll need a Raspberry Pi with Raspbian installed on the SD card for each TalkiePi you want to build. The Raspberry Pi 3 is perfect for this project as it has integrated Wi-Fi. The microphone originally used in the projects is US Robotics' USB Speakerphone (USR9610), which will be removed »

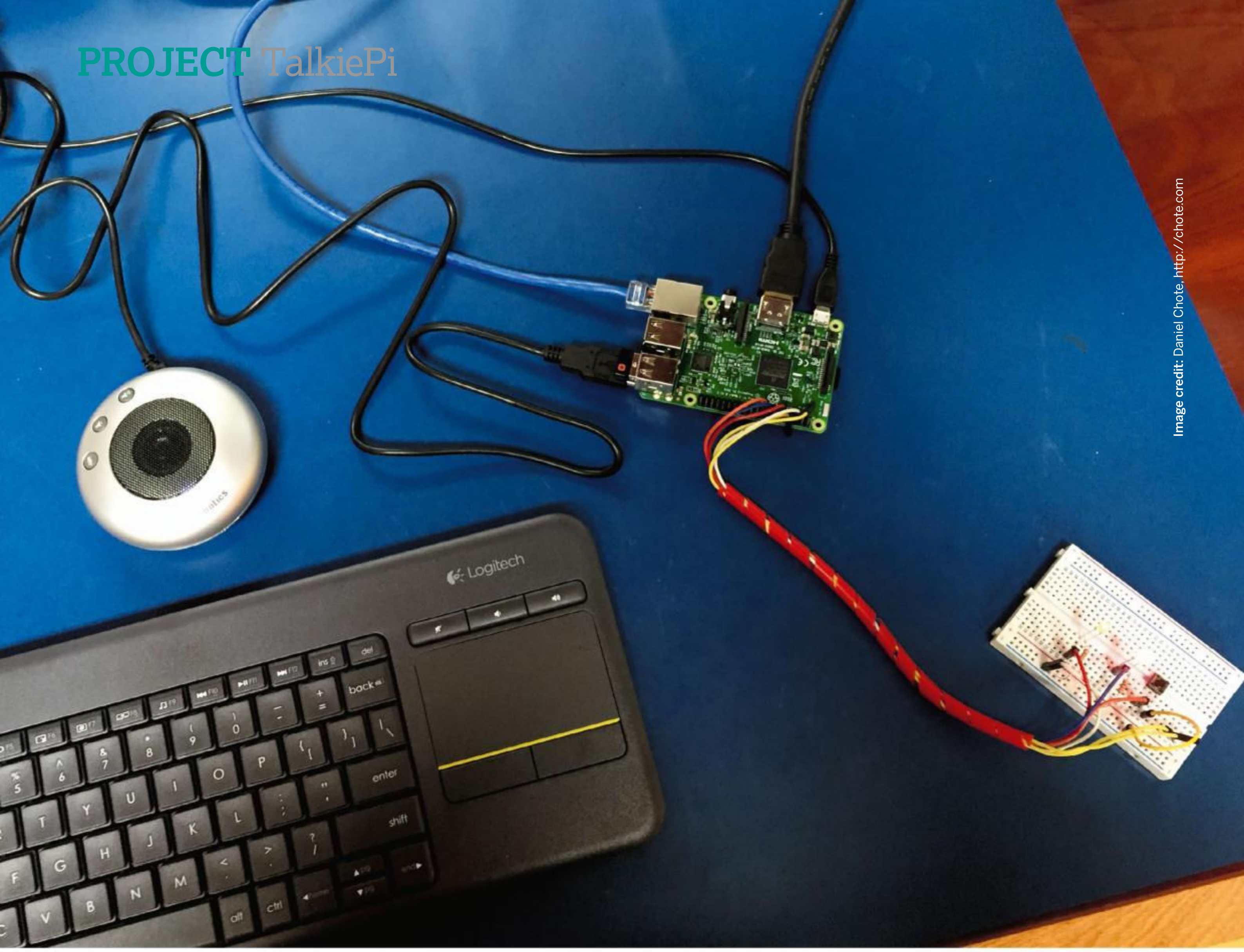


Image credit: Daniel Chote, http://chote.com

Before you start taking your speakerphone apart or soldering things together, try assembling the parts on a breadboard first to check that everything works.

from its casing and connected to the Pi. This can be hard to obtain outside the US, however, so either search eBay for the same model or see if you can find a USB speakerphone. Be careful to choose a speakerphone which works over a USB cable and not Bluetooth.

To power the Pi you'll also need a short 90 degree USB cable. (For example, Amazon UK lists a 10cm version for £1.84 from the seller SWUK.) If you're planning on printing the TalkiePi casing you'll also need five M3 nylon screws and one M3x20 nylon standoff. You can buy a pack of both of these items for £2 and £3 respectively on the Pi Hut's sister site Makersify (**https://makersify.com**). The case will also require two M3 15mm and two M3 25mm bolts. As an example, Amazon UK seller Sourcingmap sells 50-packs of each of these for £3.70 and £3.86. The speaker itself requires two M3 10mm bolts and nuts and, for example, Amazon seller Falcon Workshop Supplies Ltd offers four-packs of these for £1.40 each.

To connect your wiring and LEDs to the Pi you'll need a GPIO Header connector. Pretty much any header will do. Pi Hut, for instance, sells a 2x20 female header for £3.

Both the "status" LEDs (see below) should be 5mm, each with its own holder. Makersify sells a 50-pack of LEDs as well as a variety of Adafruit LED holders for £4-£5 each. Makersify also sells pushbutton LEDs, one of which will be necessary for the "talk" button. There are some basic light-up plastic buttons available in blue, green, red, white and yellow for £3 each. There are also more sophisticated metal pushbuttons in blue, green, red and white for £5 each. The original TalkiePi project was built with a plastic pushbutton but feel free to change this if you feel confident.

The TalkiePi also needs three 330-Ohm resistors. These should be readily available at your local electronics supplies store, but if you don't have one nearby the ever-handy Makersify sells a "basic components mixed pack" for £4, which among many other handy accessories includes 10 suitable resistors.

If you plan to use the TalkiePi as a home intercom you will most likely connect it to mains power, but if you want a truly portable transmitter consider investing in a battery pack like the PiBorg (more about this shortly).

Needless to say, if you're serious about assembling this project you'll also need wire and access to a soldering iron. The Pi Hut, for example, sells a handy Breadboarding Wire Bundle of 75 flexible stranded cord wires for just £3. For the same price, you can also invest in a half-size breadboard to experiment with.

Bear in mind that if you want to communicate with other people you will want to build two units, so you'll need double the components listed. As you'll have seen, however, most of the components come in packs so you should be able to make as many TalkiePis as you're likely to need.

Buttons and pins

The basic setup for the GPIO pins is easy to grasp. There's an LED built into the pushbutton on the front of the TalkiePi itself which lights up when you're transmitting. There are also two separate "online" and "participant" status LEDs.

If this is your first project, it's best to work through the steps on the website on your breadboard before you solder anything. Also pay close attention to the GPIO diagram on the website before you start. For full assembly instructions, visit the main project page at **http://bit.ly/Wi-FiWalkieTalkie**.

If you decide to mimic the TalkiePi project, you don't necessarily have to use a 3D-printed case, but it's certainly much handier to have trailing wires boxed up. The actual 3D designs for the casing as well as the speaker cover are based on a retro walkie-talkie, and you can find them on Daniel Chote's GitHub page **https://github.com/dchote/talkiepi/tree/master/stl**. The STL (StereoLithography) file format is compatible with most 3D printers. The TalkiePi project was done on a Monoprice Select Mini 3D printer using PLA plastic with 100 percent infill. PLA has the advantage of giving off a pleasant sugary smell when printing as well as being quite sturdy.

Bear in mind that the casing was designed specifically for the US Robotics speakerphone listed on the website, so other speakers may not fit as well. Carefully check the dimensions of your components to avoid wasting precious material.

If you already have a 3D printer you will most likely be comfortable enough from previous projects to simply produce the parts yourself. If you don't have a 3D printer, then it's best to ask a third-party company to make the top and bottom casing as well as the speaker cover for you. A simple search online for 3D print services in the UK will reveal any number of companies who are happy to make as many or as few parts as you need. The usual procedure is that you will fill in an online web enquiry form and upload the STL files to the 3D printing company's website and they will contact you with a quote. One such firm we contacted to enquire about two sets of TalkiePi top casing, bottom casing and speaker covers quoted £142.54 for manufacturing and shipping the parts! Considering that DIY 3D printers now retail for as little as £178, you may prefer to either invest in one or find a friend who will let you use theirs.

Safety in Mumbles

Mumble servers are highly customisable, and it's possible to create select groups or channels for everyone you wish to talk with. For instance, you may prefer that your friends playing World of Warcraft can't hear everyone who buzzes your front door intercom. If you do set up your Mumble Server (see *Managing Mumble*, over the page), make sure to group conversations together. You may need to use the `go run` command each time you switch channels.

To prevent your conversations being monitored, you should also consider generating an SSL certificate for your Mumble client (see *Certified Mumbler* below for information on how to do this). If you choose to install the standard Mumble client, it comes with its own certificate, but you can choose to generate another in your own name by running the certificate wizard – see **www.mumble.com/support/mumble-creating-a-certificate.php**. This enables you to generate a self-signed certificate, which is ideal for a small group of people. If you plan to open your Mumble server to the public, you may wish to obtain a trusted certificate from a bona fide certificate

»

Pi bites

If you wish to customise your TalkiePi further, PLA doesn't work well with regular paint but you can achieve good results with acrylics or nail polish. Alternatively use coloured tape.

CERTIFIED MUMBLER

Mumble software, like the TalkiePi program, can use certificates to connect to Mumble servers. This avoids the need for typing in passwords and allows you to register accounts easily with new servers. By default, this feature is switched off in TalkiePi as Daniel Chote's own server, which he set up for his children, doesn't require certificates. If you wish to use other Mumble servers, however, connect to your Raspberry Pi via SSH, then run `su mumble` to switch to the Mumble user and `cd ~` to switch to the home directory.

You'll need to generate a container for your certificate with the command:

```
openssl genrsa -aes256 -out key.pem
```

You'll also be asked to enter a password. Next create the certificate itself with the following command:

```
openssl req -new -x509 -key key.pem -out
cert.pem -days 1095
```

Enter the same password as before and when asked press Return to choose the default options such as location, e-mail address etc. Next run the following:

```
openssl rsa -in key.pem -out nopasskey.pem
```

Enter the password one last time. Finally run the following to combine your files into a single certificate:

```
cat nopasskey.pem cert.pem > mumble.pem
```

Now switch back to the root user with:

```
sudo -i
```

Then use the following command to edit Mumble's configuration:

```
nano etc/systemd/system/mumble.service
```

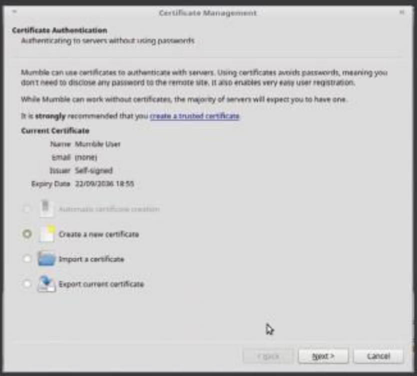
Scroll to the following line:

```
ExecStart = /home/mumble/bin/talkiepi
```

Leave a space and type the following on the same line, replacing "yourusername" with your user name:

```
-username yourusername -certificate /
home/mumble/mumble.pem
```

Press Ctrl+x, then y, then Return to save and exit. Finally reboot the Pi to effect your changes. The Raspberry Pi will now use a TLS certificate when connecting to supported Mumble servers.



Certificate Authentication
Authenticating to servers without using passwords

Mumble can use certificates to authenticate with servers. Using certificates avoids passwords, meaning you don't need to disclose any password to the remote site. It also enables very easy user registration.

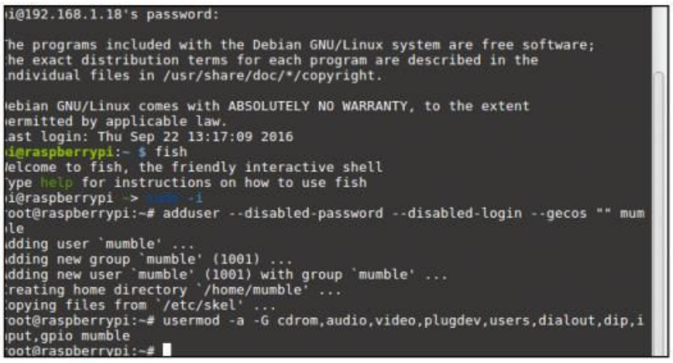
While Mumble can work without certificates, the majority of servers will expect you to have one.

It is **strongly** recommended that you create a trusted certificate.

Current Certificate
Name Mumble User
Email (none)
Issuer Self-signed
Expiry Date 22/09/2036 18:55

You can generate a certificate from the command line or alternatively if you can connect your Pi to a monitor and use the Mumble client's easy certificate wizard.

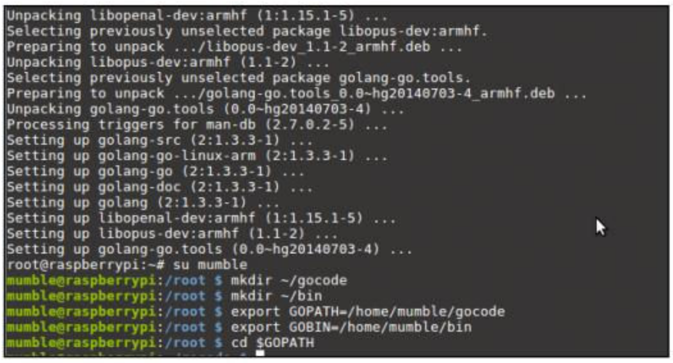
SET UP YOUR TALKIEPI



```
i@192.168.1.18's password:

he programs included with the Debian GNU/Linux system are free software;
he exact distribution terms for each program are described in the
ndividual files in /usr/share/doc/*/copyright.

ebian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
ermitted by applicable law.
ast login: Thu Sep 22 13:17:09 2016
i@raspberrypi:~ $ fish
elcome to fish, the friendly interactive shell
ype help for instructions on how to use fish
i@raspberrypi -> sudo -i
oot@raspberrypi:~# adduser --disabled-password --disabled-login --gecos "" mum
le
dding user `mumble' ...
dding new group `mumble' (1001) ...
dding new user `mumble' (1001) with group `mumble' ...
reating home directory `/home/mumble' ...
opying files from `/etc/skel' ...
oot@raspberrypi:~# usermod -a -G cdrom,audio,video,plugdev,users,dialout,dip,i
put,gpio mumble
oot@raspberrypi:~#
```

1 Create a Mumble user

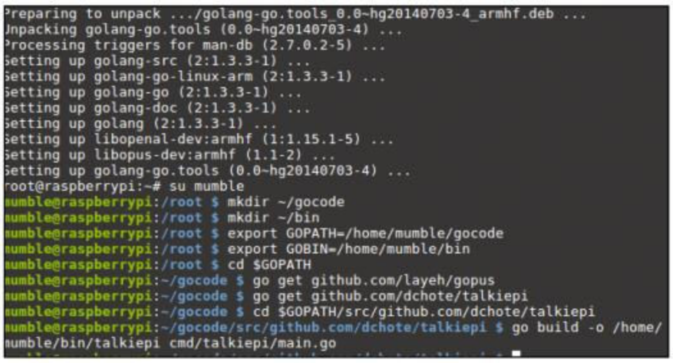
Open the Terminal on your Raspberry Pi or connect via SSH and switch to the root user with the command `sudo -i`. Next, create the user by running the following command:
`adduser --disabled-password --disabled-login --gecos "" mumble`
Give this user the necessary permissions with the following:
`usermod -a -G cdrom,audio,video,plugdev,users,dialout,dip,input,gpio mumble`



```
Unpacking libopenal-dev:armhf (1:1.15.1-5) ...
Selecting previously unselected package libopus-dev:armhf.
Preparing to unpack .../libopus-dev_1.1-2_armhf.deb ...
Unpacking libopus-dev:armhf (1.1-2) ...
Selecting previously unselected package golang-go.tools.
Preparing to unpack .../golang-go.tools_0.0~hg20140703-4_armhf.deb ...
Unpacking golang-go.tools (0.0~hg20140703-4) ...
Processing triggers for man-db (2.7.0.2-5) ...
Setting up golang-src (2:1.3.3-1) ...
Setting up golang-go-linux-arm (2:1.3.3-1) ...
Setting up golang-go (2:1.3.3-1) ...
Setting up golang-doc (2:1.3.3-1) ...
Setting up golang (2:1.3.3-1) ...
Setting up libopenal-dev:armhf (1:1.15.1-5) ...
Setting up libopus-dev:armhf (1.1-2) ...
Setting up golang-go.tools (0.0~hg20140703-4) ...
root@raspberrypi:~# su mumble
mumble@raspberrypi:/root $ mkdir ~/gocode
mumble@raspberrypi:/root $ mkdir ~/bin
mumble@raspberrypi:/root $ export GOPATH=/home/mumble/gocode
mumble@raspberrypi:/root $ export GOBIN=/home/mumble/bin
mumble@raspberrypi:/root $ cd $GOPATH
```

2 Install dependencies

Switch to your new Mumble user with `su mumble`. Next create the installation directories with `mkdir ~/gocode` and `mkdir ~/bin`. You'll need to create softlinks for these two directories with the commands:
`export GOPATH=/home/mumble/gocode`
`export GOBIN=/home/mumble/bin`
Finally, move to the install directory with `cd $GOPATH`.



```
reparing to unpack .../golang-go.tools_0.0~hg20140703-4_armhf.deb ...
npacking golang-go.tools (0.0~hg20140703-4) ...
rocessing triggers for man-db (2.7.0.2-5) ...
etting up golang-src (2:1.3.3-1) ...
etting up golang-go-linux-arm (2:1.3.3-1) ...
etting up golang-go (2:1.3.3-1) ...
etting up golang-doc (2:1.3.3-1) ...
etting up golang (2:1.3.3-1) ...
etting up libopenal-dev:armhf (1:1.15.1-5) ...
etting up libopus-dev:armhf (1.1-2) ...
etting up golang-go.tools (0.0~hg20140703-4) ...
oot@raspberrypi:~# su mumble
umble@raspberrypi:/root $ mkdir ~/gocode
umble@raspberrypi:/root $ mkdir ~/bin
umble@raspberrypi:/root $ export GOPATH=/home/mumble/gocode
umble@raspberrypi:/root $ export GOBIN=/home/mumble/bin
umble@raspberrypi:/root $ cd $GOPATH
umble@raspberrypi:~/gocode $ go get github.com/layeh/gopus
umble@raspberrypi:~/gocode $ go get github.com/dchote/talkiepi
umble@raspberrypi:~/gocode $ cd $GOPATH/src/github.com/dchote/talkiepi
umble@raspberrypi:~/gocode/src/github.com/dchote/talkiepi $ go build -o /home/
umble/bin/talkiepi cmd/talkiepi/main.go
```

3 Download and build software

To download everything run `go get github.com/layeh/gopus` and `go get github.com/dchote/talkiepi`. Next, **cd** into the TalkiePi directory: `cd $GOPATH/src/github.com/dchote/talkiepi`. Now build TalkiePi:
`go build -o /home/mumble/bin/talkiepi cmd/talkiepi/main.go`
Run `sudo -i` and makeTalkiePi boot up every time the Pi restarts:
`cp/home/mumble/gocode/src/github.com/dchote/talkiepi/conf/systemd/mumble.service /etc/systemd/system/mumble.service`

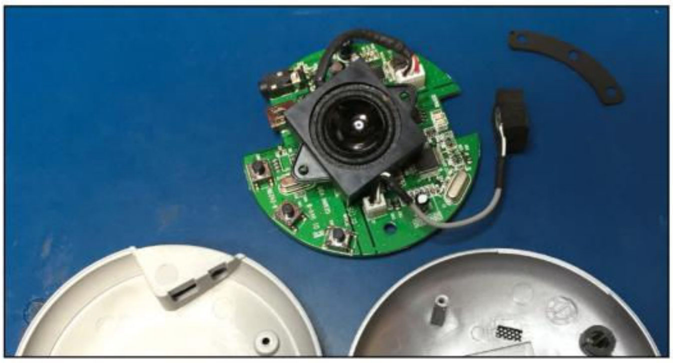


Image credit: Daniel Chote, http://chote.com

4 Connect your USB speakerphone

Connect the USB speakerphone to an available port on the Pi. At this point you'll need to refer to the instructions on the TalkiePi website at **http://bit.ly/Wi-FiWalkieTalkie** and take it stage by stage. If you print the project's case on a 3D printer, it's specifically designed to fit the PI 3 and the US Robotics speakerphone. You attach the right/left angle USB cable to the bottom-right USB port of your Pi and will need to mount the Pi and then mount the speaker to the front panel.

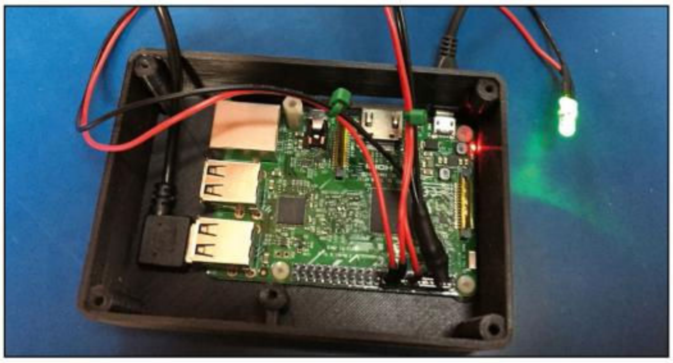
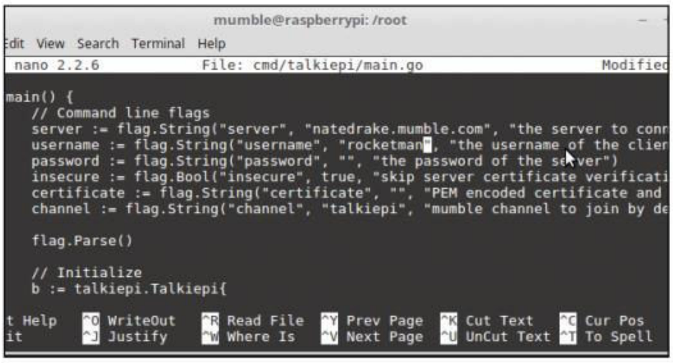


Image credit: Daniel Chote, http://chote.com

5 Optional LEDs and casing

Again, this optional stage requires following the steps on the project website at **http://bit.ly/Wi-FiWalkieTalkie**. You need to do such things as solder your positive and negative LED wires to the GPIO pins, add heat shrink to stop shorting, put together the button GPIO connector and position and fasten the LEDs. You can also place the Pi inside an enclosure with a power cable or battery pack.



```
 nano 2.2.6            File: cmd/talkiepi/main.go                   Modified

main() {
    // Command line flags
    server := flag.String("server", "natedrake.mumble.com", "the server to conn
    username := flag.String("username", "rocketman", "the username of the clien
    password := flag.String("password", "", "the password of the server")
    insecure := flag.Bool("insecure", true, "skip server certificate verificati
    certificate := flag.String("certificate", "", "PEM encoded certificate and
    channel := flag.String("channel", "talkiepi", "mumble channel to join by de

    flag.Parse()

    // Initialize
    b := talkiepi.Talkiepi{

t Help      ^O WriteOut   ^R Read File  ^Y Prev Page  ^K Cut Text   ^C Cur Pos
it          ^J Justify    ^W Where Is   ^V Next Page  ^U UnCut Text ^T To Spell
```

6 Configure Mumble Server settings

We recommend certifying your Mumble server so that you don't need to use passwords to connect, which we explain how to do in *Certified Mumbler* on the previous page. We'd also advise referring to *Managing Mumble* on the opposite page for more information on how to use your own Mumble Server and channel. You can, however, use the TalkiePi Mumble server straight away, if you wish.

authority. There are several to choose from; the Mumble wiki has a guide to doing this via Comodo (instantssl) at **http://wiki.mumble.info/wiki/Obtaining_a_Comodo_Certificate**.

By default, the TalkiePi software connects to Daniel Chote's own Mumble server and generates a username automatically for you. This keeps things simple but besides being not very fair on Chote's bandwidth, it isn't the most secure setup. See *Managing Mumble* below for how to configure TalkiePi to work with your own Mumble server.

Technically you could run the Mumble server software on the same Pi on which you're talking, but this can cause network errors, so it's best to have a separate device. If you plan to use the Pi only within the same building, consider having a machine connected to your network dedicated to running Mumble's server software, Murmur.

By default, the Pi may not use the USB speakerphone you've attached as the sound device. If you can connect your Pi to an HDMI monitor, this is fairly easy to fix by right-clicking on the volume button and adjusting your sound preferences. If you have no monitor or the Pi has already been placed in its casing, connect via SSH, and run the following commands:

```
sudo -i
aplay -l
```

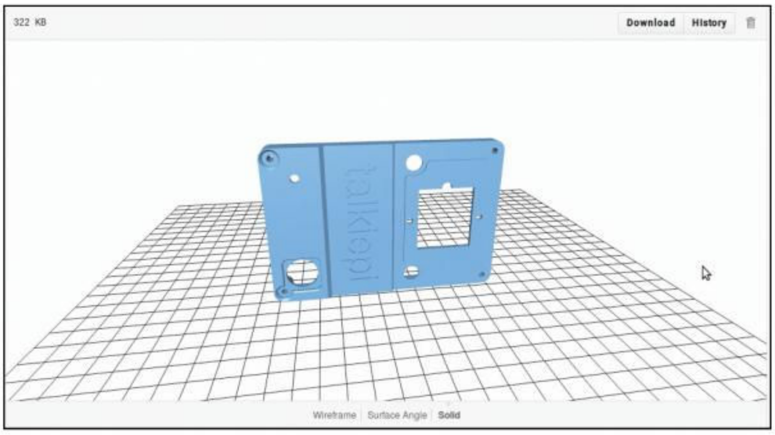
This will list all sound devices. Make a note of the "card" number of the USB device (most likely it'll be "card 1"), then run:

```
nano /usr/share/alsa/alsa.conf
```

Scroll down and change `defaults.ctl.card 0` to the corresponding number of your speakerphone. Do the same for `defaults.pcm.card 0`.

Cross Talk

The default setup for the TalkiePi is that you push the button to talk, as with old style walkie-talkies, but Mumble doesn't have to work this way. If you want to use it with a keyboard shortcut or with a headset when you raise your voice, the easiest way is to run the graphical audio wizard with the official Mumble client. Connect your Pi to a monitor, open the Terminal and run `sudo apt-get install mumble`. Next run `Mumble` and you'll be asked to configure your microphone and sound preferences. If you've set up your own Mumble server you can also use the handy menus to input the details here.



The TalkiePi project's GitHub page has handy schematics for the case. Feel free to print these with a 3D printer or make your own.

Image credit: Daniel Chote, http://chote.com

If nostalgia wins out and you decide you prefer an authentic looking walkie-talkie, bear in mind that it will be effective only while connected to the same wireless network on which the Pi was set up. This can be an issue if you need to move the Pi elsewhere, as it's not possible to connect the HDMI cable while it's inside the TalkiePi casing. Instead connect via SSH and run this command to edit your network settings:

```
sudo nano /etc/wpa_supplicant/wpa_supplicant.conf
```

At the very bottom of the file, paste the details of the new wireless network as follows:

```
network={
ssid="yourwifinetworkname"
psk="yourwifipassword"
}
```

Press Ctrl+x, then y, then Return to save and exit.

The TalkiePi is only as good as the battery pack powering it. You can either use a USB battery pack as outlined on the TalkiePi website or if you prefer a neater solution, you may want to consider using a PiBorg (**https://thepihut.com/products/battborg?variant=1103793752**). This is a custom power converter for the Pi, which works with most batteries and even comes with an eight-battery AA pack. The Pi Hut website recommends using rechargeable batteries.

The TalkiePi project is a work in progress. Currently, Chote is working on running it on a Pi Zero, which would be much smaller and less power hungry. For the latest updates see **https://github.com/dchote/talkiepi**. ■

Pi bites

For a guide for setting up Murmur for Windows and Linux machines go to **http://bit.ly/Murmur-Server**

MANAGING MUMBLE

In order to make sure your setup is secure you'll need your own Mumble server. You can set this up yourself on a separate Raspberry Pi if you plan on communicating only over your home wireless network, for example for an intercom. Alternatively, you can rent a Mumble server online. If you want to play around with a Mumble server of your own you can get a free one-month trial at **www.mumble.com/free-mumble-server.php**.

Once your server is set up, note down the domain name and its port. You should also think of a username you'd like to register and set up a channel specifically for you and your friends. (The default is Root.) Next connect to your Pi via SSH or open the Terminal and run the following command to edit TalkiePi's configuration:

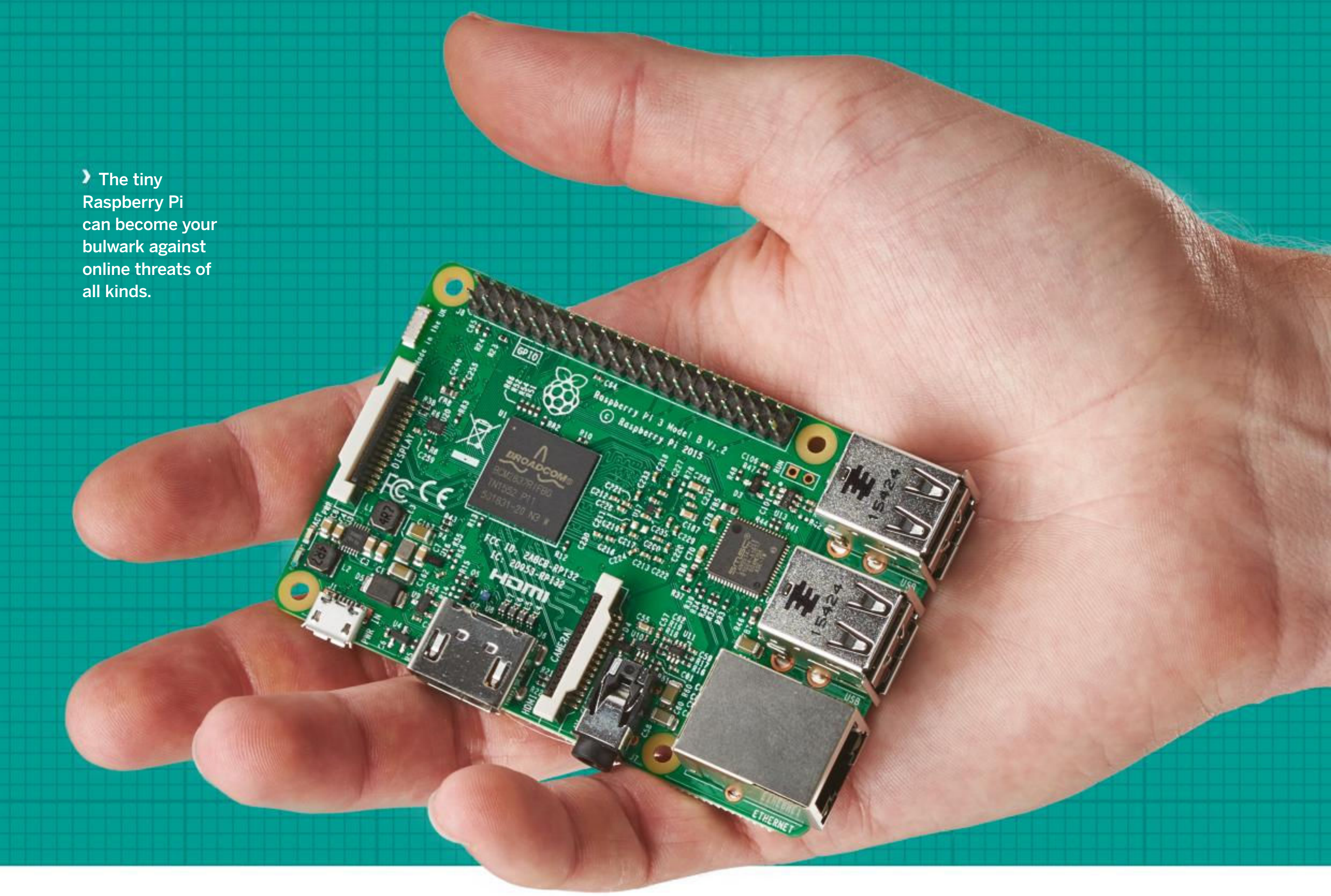
```
sudo nano /home/mumble/gocode/src/github.com/dchote/talkiepi/cmd/talkiepi/main.go
```

Scroll down to the line beginning **server** and replace the TalkiePi server address **talkiepi.projectable.me:64738** with your own, e.g. natedrake.mumble.com:64248.

On the line beginning **username** you may wish to insert a permanent username between the quotation marks. Enter a password only if your Mumble server requires one.

If you have previously created a certificate, find the line starting **insecure** and change **true** to **false**. Press Ctrl+x, then y, then Return to save your changes. Finally to start the TalkiePi software with your new settings, run the following:

```
go run /home/mumble/gocode/src/github.com/dchote/talkiepi/cmd/talkiepi/main.go
```



› The tiny Raspberry Pi can become your bulwark against online threats of all kinds.

Protect yourself with a Pi-based firewall

Turn your Pi into an ultra-secure access point complete with firewall to protect your computer from the dark side of the internet

You probably remember the news. In December 2016, the BBC reported that the infamous Mirai worm had infected a substantial number of routers used by customers of UK ISP TalkTalk, stealing their Wi-Fi passwords. The worm exploited the default admin passwords that are hardwired into a number of machines, and although the focus was on enslaving devices to take part in DOS (Denial of Service) attacks, the implications are staggering. Even those handful of users who take the trouble to change their router's default passwords seldom bother to use firewalls.

In plain English, this means your router's settings, and theoretically devices on your home wireless network, could be laid bare by attackers. Fortunately the Raspberry Pi comes to the rescue once again. We'll show you how.

This project can be roughly divided into two parts. The first is to set up your Pi as a wireless AP (access point). In other words, the Pi itself will create a wireless network for you to connect to. The Pi can then be connected to your router, so that anyone connected to its wireless network can continue to access the internet.

The second part of this project involves installing an easy-to-use firewall on your Pi which will block any open ports that attackers can exploit. You can also configure the firewall to allow access to legitimate programs, as well as block any websites you think are unsafe. Provided you have the correct equipment, you can have your own secure AP with firewall up and running in under 20 minutes.

The bare necessities

The wireless device you use must be compatible with hostapd (more about this shortly). This project is technically compatible with all models of Pi, although the Pi needs to be both wireless-enabled and have an Ethernet port to connect to your router. The Raspberry Pi 3 lends itself very well to this. Setup if at all possible should be done using a keyboard and monitor rather than SSH as once the Pi has created its own AP, you won't be able to connect to it wirelessly. You can however connect to it through your router's wireless network if you know the IP address.

If you prefer the compact Pi Zero, make sure you have both a USB OTG Host cable or a USB to MicroUSB OTG shim (available from all Pi accessory suppliers – from The Pi Hut, for example, it's just £2, **https://thepihut.com/products/usb-to-microusb-otg-converter-shim**). You'll also need an RJ45 to USB converter to attach to the USB port. You'll then be able to connect the Pi Zero to your router.

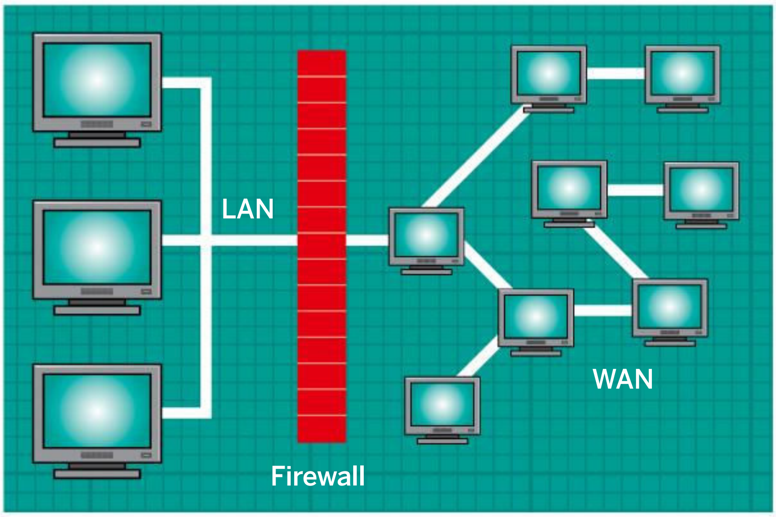
In addition to the Pi itself, you'll need a secondary device to check that the wireless AP is functioning correctly, as well as a network cable to link the Pi to the router. Most modern routers have at least one USB port, so you should be able to power the Pi from there.

For security reasons and to save on resources, it's best to have a dedicated Pi for your home firewall, rather than one that you might use for other purposes in addition. If at all possible, plug your Pi into a monitor and work from Terminal directly when going through this project rather than connect via SSH. This reduces the chance that your connection will drop at a vital stage in the proceedings.

As always, it's best to start with a clean install of the latest version of Raspbian on the Pi. Once the install is complete, open Terminal and run the following commands to be certain your system is fully up to date:

```
sudo apt-get update
sudo apt-get upgrade
```

If you do not have access to a monitor, you can connect to the Pi wirelessly through your router's Wi-Fi network via SSH. Connect your Pi to the router via Ethernet cable first. If you have a mobile device, the handy app Fing can be used to easily identify the IP address of all devices on your local network. Fing is available to download free of charge from both the iOS App Store and Google Play.



A diagram of how a firewall sits between your home network (LAN) and the internet (WAN). Individual "bricks" can be removed to let legitimate traffic through.

Once you're satisfied that the AP is working as you'd like, make sure to disable the wireless function on your router to prevent people from connecting to it instead of the Pi firewall.

Talking points

The access point is set up and managed by the handy program hostapd. Step 3 of our guide on page 69 covers creating a configuration file wherein you'll specify the settings for the Pi's new wireless network such as name, password and encryption type used. You can return to this hostapd.conf file at any time to change your settings if you wish.

»

Pi bites

To find out if your wireless card supports AP mode, open a Terminal and run the command `iw list`. If it does, you should see the initials "AP" under "valid interface combinations".

DOMAIN DENIAL

Its full name, Uncomplicated Firewall, might sound modest, but Ufw is a powerful firewall and can be used to block access to certain websites. Although technically you can also do this with dnsmasq, this would be a form of "DNS hijacking", which is a rather dastardly procedure usually carried out by scammers.

The advantage of using Ufw is that it blocks domains by IP address. This means you have to write down only one rule per address rather than remember the various domains and subdomains used by sites. Imagine trying to block every variant of Facebook.com, for instance – fb.com, m.facebook.com… the list goes on.

It stands to reason, then, that in order to block websites by IP address you'll need that information. Open Terminal on your Pi or connect via SSH and use the `host` command to view an objectionable page's IP address – for instance:

```
host strawberryfunk.com
```

This will output the IP address of the site in question. Next, use `ufw deny` to block this domain – for example:

```
sudo ufw deny out from any to 205.178.189.29
```

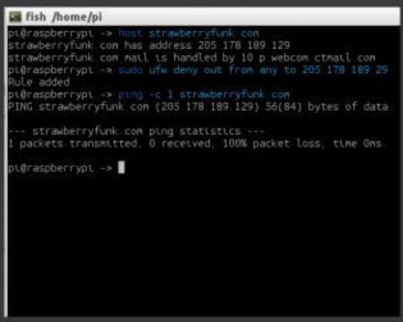
You can check that the domain has successfully been blocked by using `ping` – for example:

```
ping -c 1 strawberryfunk.com
```

If you have set up Ufw correctly, ping will report that the domain is unreachable.

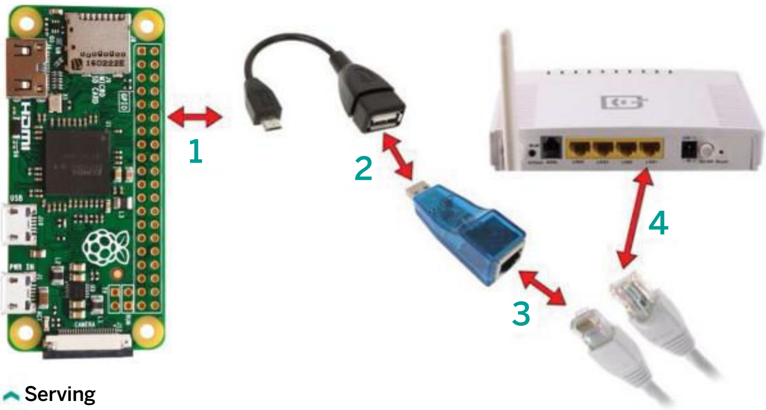
Ufw can also block outgoing connections to certain ports. For instance if you want to reduce the chance of your Pi being used to send spam messages, you can disable Port 25 with the following:

```
sudo ufw deny out 25
```



```
pi@raspberrypi -> host strawberryfunk.com
strawberryfunk.com has address 205.178.189.129
strawberryfunk.com mail is handled by 10 p.webcom.ctmail.com
pi@raspberrypi -> sudo ufw deny out from any to 205.178.189.29
Rule added
pi@raspberrypi -> ping -c 1 strawberryfunk.com
PING strawberryfunk.com (205.178.189.129) 56(84) bytes of data.

--- strawberryfunk.com ping statistics ---
1 packets transmitted, 0 received, 100% packet loss, time 0ms

pi@raspberrypi -> █
```

Use ping to check that a domain has been successfully blocked. Some websites have more than one IP address, so block each of these in turn.



Serving suggestion for Pi Zero internet connection. The OTG cable (2) allows connection of regular USB devices. The RJ45 converter (3) allows connection via Ethernet.

Pi bites

For a full rundown of Ufw's features see **https://help.ubuntu.com/community/UFW**.

While every wireless card can connect you to a Wi-Fi network, not all Wi-Fi cards can generate their own wireless networks. Fortunately the Wi-Fi module built into the Raspberry Pi 3 does support this feature, which is a good reason to consider using one for your firewall.

We'll also install the devilishly clever program dnsmasq to handle the network infrastructure. It also draws its settings from a small configuration file, which you will set up in Step 4. The suggested settings have dnsmasq use Google for its DNS server (8.8.8.8) but you can change this to another DNS provider if you wish, such as OpenDNS (208.67.222.222). For best performance, you may want to list more than one DNS server. To do this, simply add a new line for this in the dnsmasq.conf file, for example

```
server=208.67.222.222
server=208.67.222.220
```

Uncomplicated firewall

Ufw (Uncomplicated Firewall) comes bundled with a number of Linux distros and true to its name is very simple to use. It is by no means the only firewall program out there – in fact there are entire distributions of Linux such as IPFire in the wild which are specifically designed to act as a gateway between your computer and the internet, complete with colourful windows. Why then use a rather mundane command line application? Aside from the fact that Ufw has withstood the test of time, it's very easy to install and set up. Once enabled, it will launch automatically. By default it will block all incoming connections and allow all outward ones. This can be a nuisance if you and other users on your network want to use certain applications like BitTorrent or Skype, but fortunately configuring exceptions to these rules is easy. Ufw also has a graphical companion for those who are still not entirely comfortable with the command line. See *Configuring your Firewall* below for more information.

Ufw also has the ability to block connections to specific IP addresses or ranges, so you can prevent network users from visiting certain websites – see *Domain Denial* on the previous page).

Although Ufw can be used to block individual websites, any web pages that do load will display banner advertisements and pop-ups. By far the easiest way to deal with these is for you and other users on your network to install browser extensions such as Ublock Origin and Ghostery to prevent both adverts and tracking cookies. Mobile users can also benefit from the official Adblock Browser.

If this isn't feasible, you can remove the Pi's default DHCP (Dynamic Host Configuration Protocol) server and configure dnsmasq in its place to automatically redirect known ad servers to an internal IP address. By default this will create ugly blank spaces in the middle of your web pages, but you can solve this by also installing the handy app pixelserv. Pixelserv's sole purpose in life is to »

CONFIGURING YOUR FIREWALL

Once enabled, Ufw will automatically launch on startup, blocking all incoming connections and allowing all outgoing ones. This might not be the ideal setup for you, particularly if you like to connect to your Pi via SSH or VNC. To configure Ufw, use Terminal on your Pi. Ufw recognises most services like SSH, so in most cases you can enable access using `ufw allow` – for instance:

```
sudo ufw allow ssh
```

To see a full list of services you can enable in this way, simply run the following command:

```
nano /etc/services
```

If your service isn't listed, you can simply open the port it uses. For instance RealVNC, which can be used to connect remotely to the Pi's desktop, runs on Port 5800. To have Ufw open it, run:

```
sudo ufw allow 5800
```

To allow connections only from a specific IP address, use `from` – for instance:

```
sudo ufw allow from 192.168.1.12
```

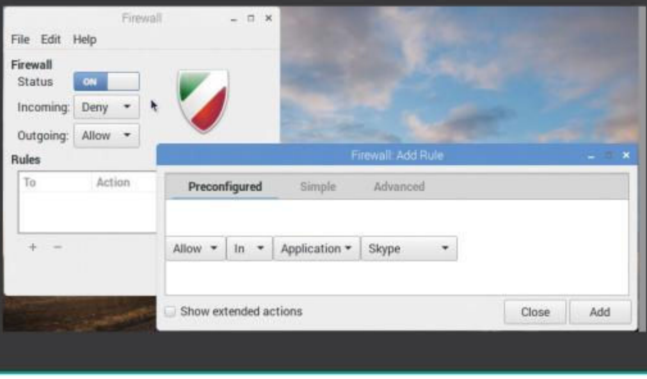
For a complete readout of the Firewall settings in Ubuntu, use the following:

```
sudo ufw status verbose
```

To make life even easier for you, Ufw has a graphical companion, Gufw. To open it, run:

```
sudo gufw
```

There's a handy rocker switch as well as a list of open and blocked ports. Click the + button to add additional rules. Click "Show extended actions" to see preconfigured settings for certain popular applications such as Skype.

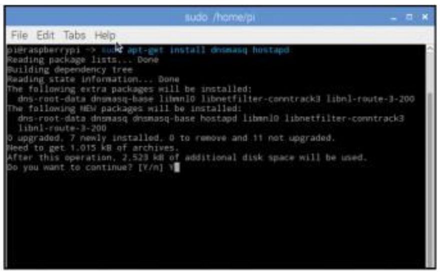


Firewall
Status: ON
Incoming: Deny
Outgoing: Allow

Allow · In · Application · Skype

Gufw is the graphical companion to Ufw. It has preconfigured rules to allow more popular apps such as Skype. This saves you the trouble of having to work out which port numbers to open.

CREATE YOUR FIREWALL WITH UFW


sudo /home/pi
File Edit Tabs Help

```
pi@raspberrypi ~ $ sudo apt-get install dnsmasq hostapd
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  dns-root-data dnsmasq-base libmnl0 libnetfilter-conntrack3 libnl-route-3-200
The following NEW packages will be installed:
  dns-root-data dnsmasq dnsmasq-base hostapd libmnl0 libnetfilter-conntrack3
  libnl-route-3-200
0 upgraded, 7 newly installed, 0 to remove and 11 not upgraded.
Need to get 1,015 kB of archives.
After this operation, 2,523 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y
```

1 Install prerequisites

Plug your Pi into the Ethernet port on your router, then connect via SSH or ideally open Terminal on the Pi. Install the necessary programs by running:

```
sudo apt-get install dnsmasq hostapd
```

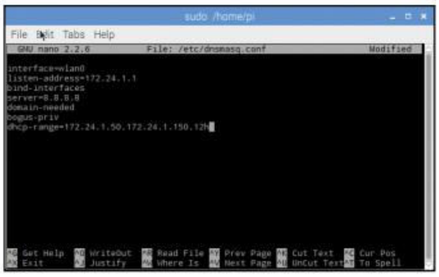
Next, run the following to open the configuration file:

```
sudo nano /etc/dhcpcd.conf
```

Add these lines to the very bottom of the file:

```
interface wlan0
static ip_address=172.24.1.1/24
```

Press Ctrl+X, Y, then Return to save and exit.


sudo /home/pi
File Edit Tabs Help
^G Get Help
^O WriteOut
^R Read File
^Y Prev Page
^K Cut Text
^C Cur Pos
^X Exit
^J Justify
^W Where Is
^V Next Page
^U UnCut Text
^T To Spell

```
  GNU nano 2.2.6              File: /etc/dnsmasq.conf                 Modified

interface=wlan0
listen-address=172.24.1.1
bind-interfaces
server=8.8.8.8
domain-needed
bogus-priv
dhcp-range=172.24.1.50,172.24.1.150,12h
```

4 Configure dnsmasq

Move the old dnsmasq configuration file with:

```
sudo mv /etc/dnsmasq.conf /etc/
dnsmasq.conf.orig
```

Then create a new one by running:

```
sudo nano /etc/dnsmasq.conf
```

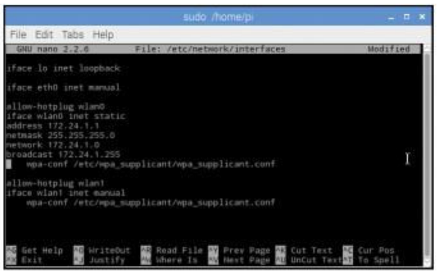
Paste in the following text:

```
interface=wlan0
listen-address=172.24.1.1
bind-interfaces     things elsewhere
server=8.8.8.8
domain-needed
bogus-priv
dhcp-range=172.24.1.50,172.24.1.150,
12h
```

Save and exit. Now run:

```
sudo nano /etc/sysctl.conf
```

Find the line starting "**net.ipv4.ip_forward=1**" and remove the # at the start. Save, exit, and reboot the Pi.


sudo /home/pi
File Edit Tabs Help
^G Get Help
^O WriteOut
^R Read File
^Y Prev Page
^K Cut Text
^C Cur Pos
^X Exit
^J Justify
^W Where Is
^V Next Page
^U UnCut Text
^T To Spell

```
  GNU nano 2.2.6         File: /etc/network/interfaces                Modified

iface lo inet loopback

iface eth0 inet manual

allow-hotplug wlan0
iface wlan0 inet static
address 172.24.1.1
netmask 255.255.255.0
network 172.24.1.0
broadcast 172.24.1.255
    wpa-conf /etc/wpa_supplicant/wpa_supplicant.conf

allow-hotplug wlan1
iface wlan1 inet manual
    wpa-conf /etc/wpa_supplicant/wpa_supplicant.conf
```

2 Set Static IP

Open your network interfaces configuration with:

```
sudo nano /etc/network/interfaces
```

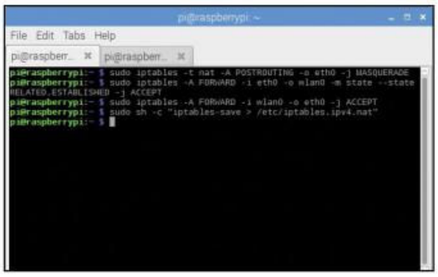
Find the line "**iface wlan0 inet static**" and change it to:

```
iface wlan0 inet manual
```

Press Return to start a new line, then paste the following:

```
address 172.24.1.1
netmask 255.255.255.0
network 172.24.1.0
broadcast 172.24.1.255
```

Place a # at the start of the line beginning "wpa-conf". Save and exit in the same way as before.


pi@raspberrypi: ~
File Edit Tabs Help
pi@raspberr...
pi@raspberr...

```
pi@raspberrypi:~ $ sudo iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
pi@raspberrypi:~ $ sudo iptables -A FORWARD -i eth0 -o wlan0 -m state --state
RELATED,ESTABLISHED -j ACCEPT
pi@raspberrypi:~ $ sudo iptables -A FORWARD -i wlan0 -o eth0 -j ACCEPT
pi@raspberrypi:~ $ sudo sh -c "iptables-save > /etc/iptables.ipv4.nat"
pi@raspberrypi:~ $
```

5 Set up IPV4 Forwarding

Open Terminal on your Pi and run:

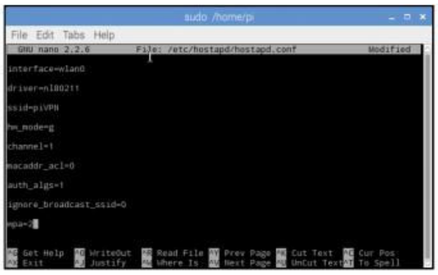
```
sudo iptables -t nat -A
POSTROUTING -o eth0 -j
MASQUERADE
sudo iptables -A FORWARD -i eth0 -o
wlan0 -m state --state
RELATED,ESTABLISHED -j ACCEPT
sudo iptables -A FORWARD -i wlan0
-o eth0 -j ACCEPT
sudo sh -c "iptables-save > /etc/
iptables.ipv4.nat"
```

Next, run:

```
sudo nano /etc/rc.local
```

Paste the following two lines just above the line reading "**exit 0**":

```
iptables-restore < /etc/iptables.ipv4.nat
/usr/sbin/hostapd /etc/hostapd/
hostapd.conf
```


sudo /home/pi
File Edit Tabs Help
^G Get Help
^O WriteOut
^R Read File
^Y Prev Page
^K Cut Text
^C Cur Pos
^X Exit
^J Justify
^W Where Is
^V Next Page
^U UnCut Text
^T To Spell

```
  GNU nano 2.2.6         File: /etc/hostapd/hostapd.conf              Modified

interface=wlan0

driver=nl80211

ssid=piVPN

hw_mode=g

channel=1

macaddr_acl=0

auth_algs=1

ignore_broadcast_ssid=0

wpa=2
```

3 Set up Access Point

Open the hostapd.conf file by running

```
sudo nano /etc/hostapd/hostapd.conf
```

Paste the following:

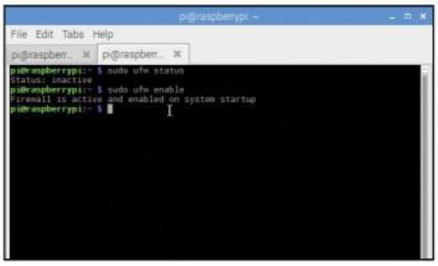
```
interface=wlan0
driver=nl80211
ssid=piVPN
hw_mode=g
channel=1
macaddr_acl=0
auth_algs=1
ignore_broadcast_ssid=0
wpa=2
wpa_key_mgmt=WPA-PSK
wpa_passphrase=raspberry231
wpa_pairwise=TKIP
rsn_pairwise=CCMP
```

Change the SSID, passphrase and network encryption as you see fit. Next run:

```
sudo nano /etc/default/hostapd
```

Find the line starting **#DAEMON_CONF=""**. Remove the # at the start of the line and change it to:

```
DAEMON_CONF="/etc/hostapd/
hostapd.conf"
```


pi@raspberrypi: ~
File Edit Tabs Help
pi@raspberr...
pi@raspberr...

```
pi@raspberrypi:~ $ sudo ufw status
Status: inactive
pi@raspberrypi:~ $ sudo ufw enable
Firewall is active and enabled on system startup
pi@raspberrypi:~ $
```

6 Set up Firewall

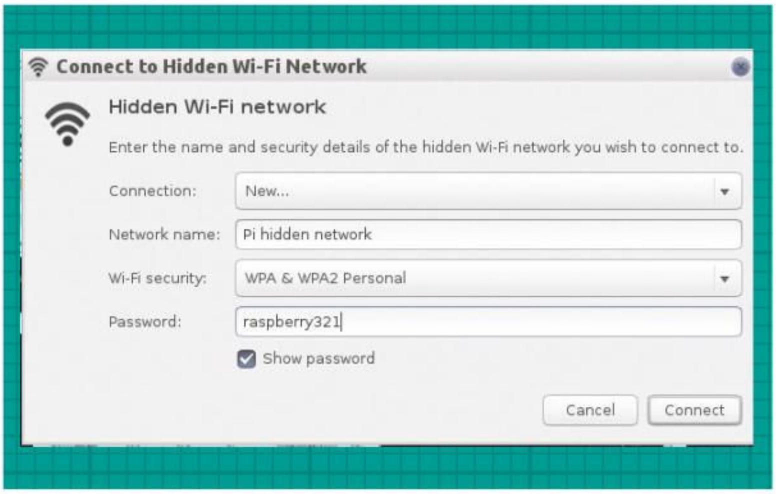
Run the following commands, one after the other:

```
sudo update-rc.d hostapd enable
sudo update-rc.d dnsmasq enable
```

Reboot the Pi, and reopen Terminal. To install and then enable the firewall run the following:

```
sudo apt-get install ufw gufw
sudo ufw enable
```

You may wish to configure the firewall at this stage – see *Configuring your Firewall* on the opposite page.


Connect to Hidden Wi-Fi Network

Hidden Wi-Fi network

Enter the name and security details of the hidden Wi-Fi network you wish to connect to.

Connection: New...

Network name: Pi hidden network

Wi-Fi security: WPA & WPA2 Personal

Password: raspberry321

☑ Show password

Connecting to a hidden network makes life only slightly more difficult for other network users but much harder for hackers, as they need both the network name and password.

support ad-blocking via serving a transparent 1x1 pixel GIF image in place of adverts. This makes your ad-blocking a much tidier experience.

If you're interested in adding an ad-blocking feature to the Pi, a tutorial is available on the Adafruit website: **https://learn.adafruit.com/raspberry-pi-as-an-ad-blocking-access-point**

Building bridges

This project focuses on bridging the connection between your Pi's wireless interface and the Ethernet interface. This allows your Access Point to remain connected to the internet. If you need to double-check the names of your interfaces, you can run the command `ifconfig` at any time. By default the names of these interfaces should be "wlan0" and "eth0" respectively.

If you're unable to plug the Pi into a router, theoretically you could add another wireless interface such as the official Raspberry Pi Wi-Fi module and use it to access your router's wireless network. However, this is less secure and will reduce your connection speed. If your router is somewhere hard to reach (or actually out of range), consider using a "homeplug" style device which uses the power lines in your home or office for network connections.

Do not despair if the Access Point or the firewall do not work the first time you go through the steps. Fortunately the Pi's network settings are very flexible, so you can usually go back over the steps again without having to reinstall Raspbian.

If you're using a Wi-Fi driver besides that built into the Pi 3 or the official Raspberry Pi Wi-Fi adapter then you may have to change the "**driver=**" value in **hostapd.conf**. For information visit **http://linuxwireless.org/en/users/Documentation/hostapd**.

You can disable the Ufw firewall permanently and delete all rules with the following command:

```
sudo ufw reset
```

If required, you can also restore the settings you previously backed up with dnsmasq by reversing the original command:

```
sudo mv /etc/dnsmasq.conf.orig /etc/dnsmasq.conf
```

If you choose to block specific ports or services (see *Domain Denial* on page 67), bear in mind that IT-literate users on your network may bypass this via a VPN. You can of course use Ufw to block the ports commonly used by VPNs, such as 1154, or block all outgoing traffic and then enable it for specific applications and services. Some applications will randomise the ports used, making it difficult to lock down specific protocols like BitTorrent. You might consider disabling UDP (User Datagram Protocol) if you want to prevent streaming of most music and video sites.

If you choose to use a browser extension or pixelserv to block ads, certain sites may display incorrectly or fail to load altogether. Adblocking extensions can usually be disabled temporarily by clicking on their icon in your browser's menu bar.

For security reasons, it would be wise to use a dedicated Pi exclusively as an Access Point and avoid placing any personal data on it. For extra safety, make sure to use a long, robust password – the one in the tutorial is only by way of example.

For the uber-cautious, consider creating a hidden Wi-Fi network for the Pi. Edit the hostapd configuration by opening a Terminal and typing:

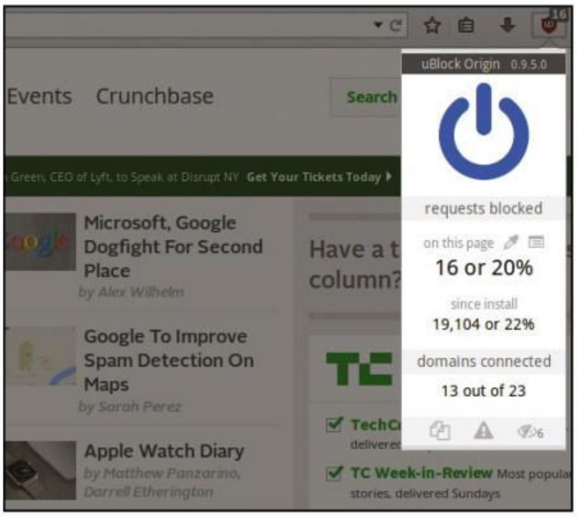
```
/etc/hostapd/hostapd.conf
```

Change "**ignore_broadcast_ssid=0**" to "**ignore_broadcast_ssid=1**".

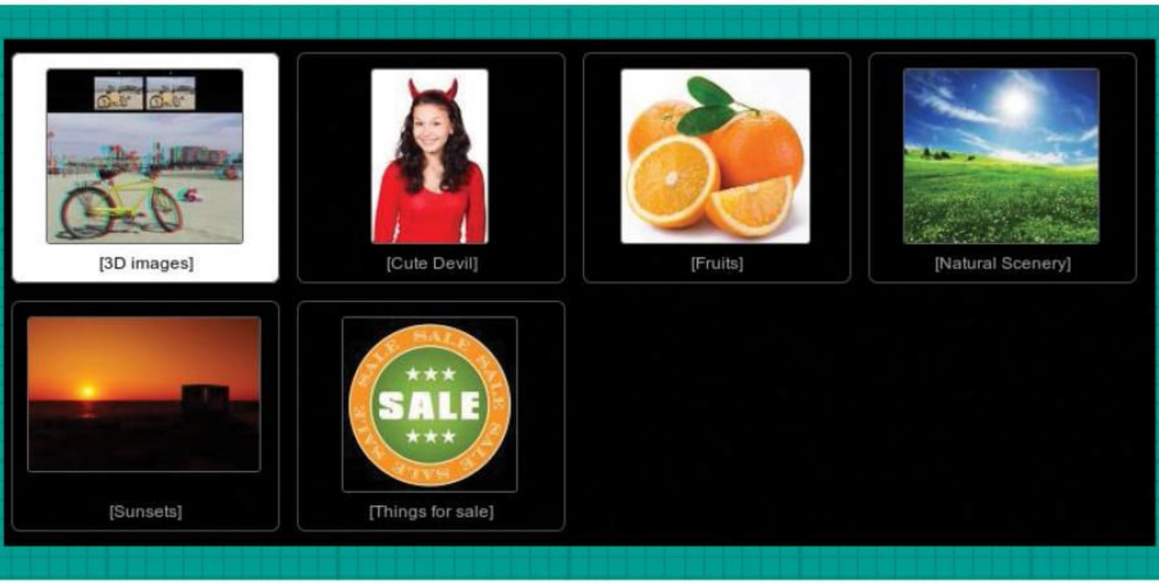
From now on all devices will have to enter the name of the network as well as the password in order to connect. Consider setting a new name when editing the file to be extra secure.

If you have devices using a cabled connection like a home server, you can continue connecting these directly to your router for internet access but bear in mind that they won't be visible on the Pi's wireless network. This works both ways, however – the Pi will not interfere with their operations at all.

Finally, bear in mind that your router may also use a firewall. If you're comfortable with configuring your router, you can choose to open and close ports to match Ufw's settings. Alternatively you can disable the router's firewall altogether and let the Pi manage everything. ■


Events
Crunchbase
Search
Green, CEO of Lyft, to Speak at Disrupt NY Get Your Tickets Today
Microsoft, Google Dogfight For Second Place
by Alex Wilhelm
Google To Improve Spam Detection On Maps
by Sarah Perez
Apple Watch Diary
by Matthew Panzarino, Darrell Etherington
Have a t column?
TC
uBlock Origin 0.9.5.0
requests blocked
on this page
16 or 20%
since install
19,104 or 22%
domains connected
13 out of 23
TC Week-in-Review Most popular stories, delivered Sundays

The handy browser add-on uBlock Origin in action. You can click on the unobtrusive icon at any time to temporarily disable ad-blocking if you wish to do so on a particular page or see which sites have been blocked.



While pictures of your favourite fruits and Turner-esque sunsets are innocuous enough, you may prefer to be certain that pictures of your loved ones stay private.

Build a Pi-based photo gallery

We show you how to view, collate and organise your favourite snaps by setting up an easy photo server on the Raspberry Pi

These days most of us have thousands of digital photos, usually scattered over hard disks or removable media of some kind, just crying out to be organised, viewed and enjoyed instead of languishing in digital limbo. SFPG (Single File PHP Gallery), the creation of Danish developer Kenny Svalgaard, enables you to set up your very own photo server within minutes, capable of displaying your photos, organising them into albums, running slideshows and much more.

In this project we will explore first how to set up your Pi as a server, using the light and open source software Lighttpd (pronounced “lighty”). Once the server is up and running, it’s a simple matter of setting up the necessary folders, then downloading and extracting SFPG. It’s not every day that you can become a server administrator and organise your digital life with three easy steps but here at Pi User we aim to please.

This project is compatible with all models of Raspberry Pi. Before proceeding, note that your images will be stored on the Pi’s SD card, so either make sure you have sufficient space or use the Pi’s own SD Card Copier utility to move your data to a card with larger capacity.

We’ll be using the most recent version of SFPG (4.7.0), which was released in December 2016. We recommend you check the Downloads section of the developer’s website (**http://sye.dk/sfpg**) before you begin, to make sure the links are right.

Can’t I just use...?

From Google Photos to Instagram to Facebook, the number of websites offering to collate and sort your most personal pictures is as boundless as the sea, to paraphrase Shakespeare. So why add the drop that is SFPG to the vast ocean of photo gallery software? It doesn’t even come with a handy mobile app or the ability to place a dog’s nose and ears over your face.

Well, the most compelling reason for using SFPG is that it is truly cross-platform – all you need to view and browse your photos is a web browser. Unlike other self-hosting image solutions, it also doesn’t require you to spend ages setting up an SQL database to store all your pictures. With its default configuration SFPG makes photos on the Pi available only to your home network. This makes it perfect for families, schools, workplaces or indeed any situation where not everyone needs to see that picture of you or one of your loved

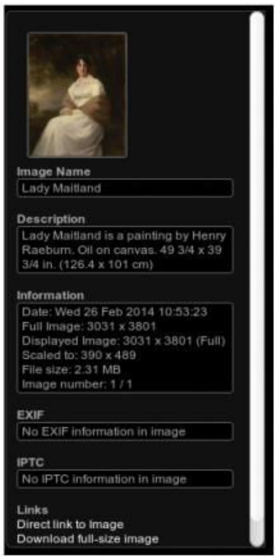


Image Name
Lady Maitland

Description
Lady Maitland is a painting by Henry Raeburn. Oil on canvas. 49 3/4 x 39 3/4 in. (126.4 x 101 cm)

Information
Date: Wed 26 Feb 2014 10:53:23
Full Image: 3031 x 3801
Displayed Image: 3031 x 3801 (Full)
Scaled to: 390 x 489
File size: 2.31 MB
Image number: 1 / 1

EXIF
No EXIF information in image

IPTC
No IPTC information in image

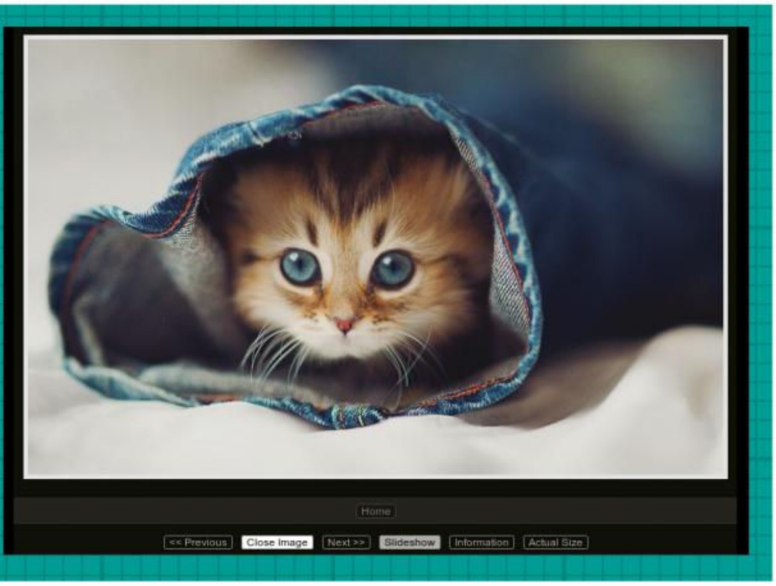
Links
Direct link to image
Download full-size image

Click the Information button to see the lowdown on individual images. You can add a description using Admin Mode.

Pi bites

To see a quick demonstration of what SFPG can do, visit **https://sye.dk/sfpg/demo**. Remember that loading times will not be as fast over the web as when you’re accessing photos on your Pi’s home network.

»



SFPG supports a slideshow mode so you can cycle your pictures of fluffy kittens according to your mood.

Pi bites

If your photos are located on another device such as a laptop, enable Admin mode to create new albums and upload them via your web browser (see the "Admin Mode" box for details).

ones spilling out of a taxi last Friday night wearing a pirate hat. Naturally many third-party photo hosts have strict privacy policies and offer settings to limit the audience for your pictures. However these are only as useful as the company's ability to protect themselves from hackers, subpoenas and users who haven't configured their privacy settings properly.

Finally, SFPG is extremely customisable. You can change the way the date appears, add watermarks, change the size and appearance of image thumbnails... and there's no limit on how many images you can store – the only restriction is the capacity of your SD card. Hardened coders may also appreciate the ability to customise the default index.php file to use backgrounds and colours of their own choosing.

Moving photos

Once SFPG is up and running, you'll need to copy your pictures to the **photos** folder in **/var/www/html/**. If the photos are already inside a folder, rename it to something meaningful, such as "Planes", so you won't become confused. If you've followed the steps in the tutorial on the opposite page, you can Copy and Paste using Ctrl+C and Ctrl+V between your Pictures folder and the photos folder.

If you're connecting to your Pi via SSH or just want to avoid repeatedly copying and pasting, you can move files or folders with the `cp` command. For instance to copy the folder "Planes" in your Pi's Pictures folder over to SFPG's photos folder, use the following:

```
cp -a /home/pi/Pictures/Planes /var/www/html/
photos/planes/
```

SFPG will automatically create a gallery album using the folder name and list the photos inside.

If your gallery fails to load at all, double-check that lighttpd server is running by opening a web browser on another device connected to your network and going to **http://ipaddressofyourpi** (for example, http://192.168.43.87 – to find your Pi's IP address, simply run the command `ifconfig` on your Pi). You should see a placeholder page (see opposite page). If not, work through the first step of the tutorial again.

The default viewing mode for Gallery doesn't allow you to alter the photos or upload new ones from your web browser. Enable admin mode to do this (see the *Admin Mode* box below).

If changes you make in Admin Mode aren't taking effect, you most likely have a permissions issue. You can run the following command on the Pi to fix this:

```
sudo chmod -R 777 /var/www/html/photos
```

Bear in mind that this will give read/write permissions to the photos folder to any user on your network

If you followed the steps in the tutorial opposite, then your Pi is now working as a web server, so there's no reason why others can't connect to the photo gallery remotely. You will however need to

Enabling Admin Mode in SFPG

By default Single File PHP Gallery displays only content, and the only way to create new albums, add and remove photos and edit their metadata is by working directly with the photos on the Pi.

Fortunately SFPG can be run in Admin mode. To do this, first open Terminal on the Pi or connect remotely via SSH and edit the index file by running:

```
sudo nano /var/www/html/photos/index.php
```

Scroll down to the text **define('ADMIN', FALSE);** and change "FALSE" to "TRUE". Press Ctrl + X, Y, then Enter to save and exit.

Reload the page with your photo gallery if it's already open. The Admin button will now appear on the right-hand side. When viewing albums or photos, first click the Admin button so it's highlighted. Next select an individual album or photo. A red border will appear around the selected item. You can now rename the file, move it, delete it and more. You can also use the Upload button to copy one or more files into the Gallery. Hold down Ctrl and click to select multiple photos.

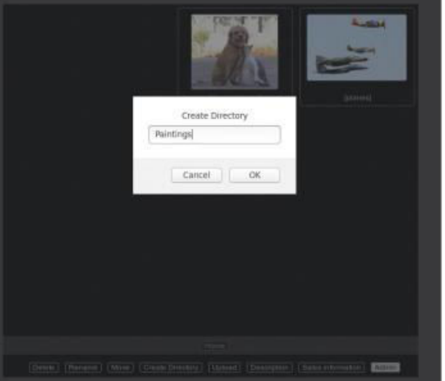
With this setup, anyone connected to your network can both view and modify your images. For extra peace of mind, you can set a password to your photo gallery. Run the command below once again:

```
sudo nano /var/www/html/photos/index.php
```

Scroll down to **define('PASSWORD', '');**. You'll find this just above the value you changed earlier to enable Admin mode. Insert the password of your choice within the quotation marks – for instance:

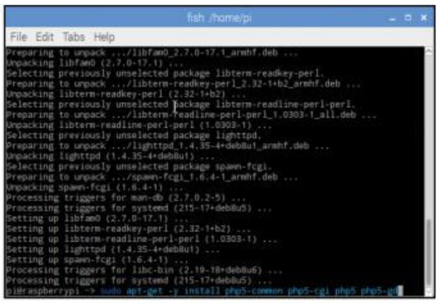
```
define('PASSWORD', 'password123');
```

Alternatively if you wish to set up a Public Gallery with Admin access for select users, follow the steps outlined in the **readme.txt** file in **/var/www/html/photos**.



One use of the admin tool: you can add new albums by clicking Admin, then Create Directory. Click on the new album, then on Upload to add images.

INSTALL AND SET UP SFPG



```
Preparing to unpack .../libfam0_2.7.0-17.1_armhf.deb ...
Unpacking libfam0 (2.7.0-17.1) ...
Selecting previously unselected package libterm-readkey-perl.
Preparing to unpack .../libterm-readkey-perl_2.32-1+b2_armhf.deb ...
Unpacking libterm-readkey-perl (2.32-1+b2) ...
Selecting previously unselected package libterm-readline-perl-perl.
Preparing to unpack .../libterm-readline-perl-perl_1.0303-1_all.deb ...
Unpacking libterm-readline-perl-perl (1.0303-1) ...
Selecting previously unselected package lighttpd.
Preparing to unpack .../lighttpd_1.4.35-4+deb8u1_armhf.deb ...
Unpacking lighttpd (1.4.35-4+deb8u1) ...
Selecting previously unselected package spawn-fcgi.
Preparing to unpack .../spawn-fcgi_1.6.4-1_armhf.deb ...
Unpacking spawn-fcgi (1.6.4-1) ...
Processing triggers for man-db (2.7.0.2-5) ...
Processing triggers for systemd (215-17+deb8u5) ...
Setting up libfam0 (2.7.0-17.1) ...
Setting up libterm-readkey-perl (2.32-1+b2) ...
Setting up libterm-readline-perl-perl (1.0303-1) ...
Setting up lighttpd (1.4.35-4+deb8u1) ...
Setting up spawn-fcgi (1.6.4-1) ...
Processing triggers for libc-bin (2.19-18+deb8u6) ...
Processing triggers for systemd (215-17+deb8u5) ...
pi@raspberrypi -> sudo apt-get -y install php5-common php5-cgi php5 php5-gd
```

1 Install server software

Open Terminal on your Pi or connect via SSH and first update your system with the usual command:

```
sudo apt-get update
```

Next install Lighttpd with:

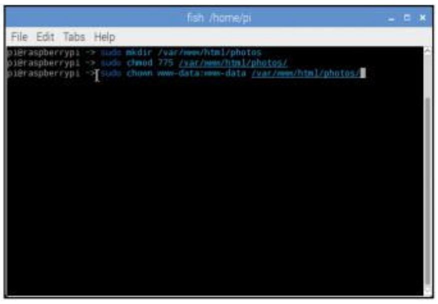
```
sudo apt-get -y install lighttpd
```

Install the required PHP files with

```
sudo apt-get -y install php5-common
php5-cgi php5 php5-gd
```

Next, enable the Fast CGI module with

```
sudo lighty-enable-mod fastcgi-php
```



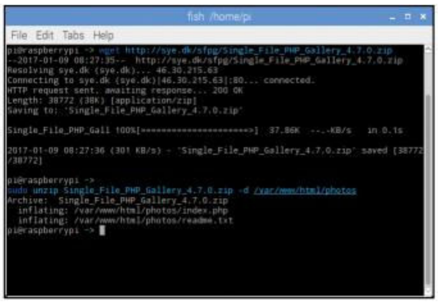
```
pi@raspberrypi -> sudo mkdir /var/www/html/photos
pi@raspberrypi -> sudo chmod 775 /var/www/html/photos/
pi@raspberrypi -> sudo chown www-data:www-data /var/www/html/photos/
```

2 Modify Folder Permissions and reboot

Create the server's directories, then allow yourself write access to them, by running the following commands:

```
sudo mkdir /var/www/html/photos
sudo chown www-data:www-data /
var/www
sudo chmod -R 775 /var/www
sudo usermod -a -G www-data pi
```

Reboot the Pi when you are done to apply the changes. Check that the server is running by navigating to your Pi's IP address on another device connected to your network. (If you don't know your Pi's IP address, you can find it by simply running the command `ifconfig` on your Pi.)



```
pi@raspberrypi -> wget http://sye.dk/sfpg/Single_File_PHP_Gallery_4.7.0.zip
--2017-01-09 08:27:35--  http://sye.dk/sfpg/Single_File_PHP_Gallery_4.7.0.zip
Resolving sye.dk (sye.dk)... 46.30.215.63
Connecting to sye.dk (sye.dk)|46.30.215.63|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 38772 (38K) [application/zip]
Saving to: 'Single_File_PHP_Gallery_4.7.0.zip'

Single_File_PHP_Gall 100%[=====================>]  37.86K  --.-KB/s   in 0.1s

2017-01-09 08:27:36 (301 KB/s) - 'Single_File_PHP_Gallery_4.7.0.zip' saved [38772
/38772]

pi@raspberrypi ->
sudo unzip Single_File_PHP_Gallery_4.7.0.zip -d /var/www/html/photos
Archive:  Single_File_PHP_Gallery_4.7.0.zip
  inflating: /var/www/html/photos/index.php
  inflating: /var/www/html/photos/readme.txt
pi@raspberrypi ->
```

3 Install SFPG

Open Terminal again and run `wget` to download the current version of the SFPG software – at the time of writing, this is as follows:

```
wget http://sye.dk/sfpg/Single_File_
PHP_Gallery_4.7.0.zip
```

Extract it to your new **photos** folder with `unzip` – for example:

```
sudo unzip Single_File_PHP_
Gallery_4.7.0.zip -d /var/www/html/
photos
```

Now run the following:

```
sudo nano /var/www/html/photos/
index.php
```

Scroll down to **define('SECURITY_PHRASE', '');**. Enter a random string here, for example **define('SECURITY_PHRASE', 'Dj5remJlpQ');**

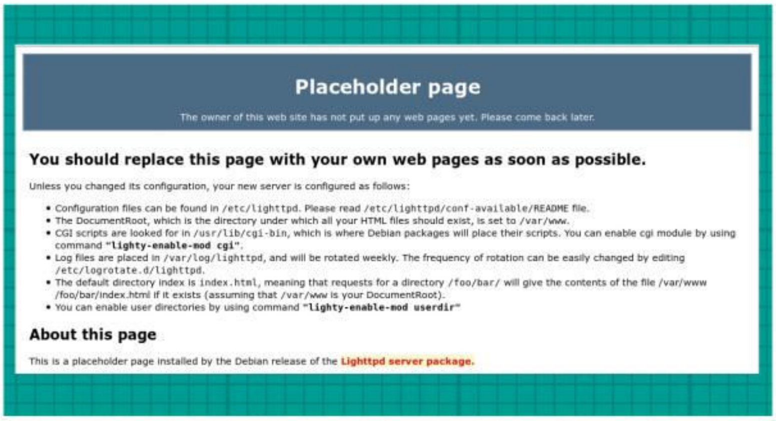
Move your pictures into **/var/www/html/photos**. To view them, go to **http://ipaddressofyourpi/photos** in your browser.

forward ports on your router so that others can connect to the Pi outside your home network. You will need to find your Pi's IP address, as already explained, then use this to enable port forwarding. The steps for port forwarding will vary from router to router, but you may find instructions for your specific model on **http://portforward.com**.

Remember that if the connection to your Pi is not secured via SSL then anyone monitoring your connection can view the photos, even without a password. Web browsers often cache images, so try to encourage users to view SFPG in Private or Incognito browsing mode, which won't retain any data once the web browser is closed.

Alternatively, see Lighttpd's guide at **https://redmine.lighttpd.net/projects/1/wiki/HowToSimpleSSL** to easily set up a secure connection to your server. This will create a "self-signed" certificate, so users may see a warning the first time they connect. It's safe to ignore this.

Read through the readme.txt file in **/var/www/html/photos** to view all options for configuring SFPG, such as setting the size for thumbnails or changing how the date and time are displayed.



Placeholder page

The owner of this web site has not put up any web pages yet. Please come back later.

You should replace this page with your own web pages as soon as possible.

Unless you changed its configuration, your new server is configured as follows:

- Configuration files can be found in `/etc/lighttpd`. Please read `/etc/lighttpd/conf-available/README` file.
- The DocumentRoot, which is the directory under which all your HTML files should exist, is set to `/var/www`.
- CGI scripts are looked for in `/usr/lib/cgi-bin`, which is where Debian packages will place their scripts. You can enable cgi module by using command "**lighty-enable-mod cgi**".
- Log files are placed in `/var/log/lighttpd`, and will be rotated weekly. The frequency of rotation can be easily changed by editing `/etc/logrotate.d/lighttpd`.
- The default directory index is `index.html`, meaning that requests for a directory `/foo/bar/` will give the contents of the file /var/www/foo/bar/index.html if it exists (assuming that `/var/www` is your DocumentRoot).
- You can enable user directories by using command "**lighty-enable-mod userdir**"

About this page

This is a placeholder page installed by the Debian release of the **Lighttpd server package.**

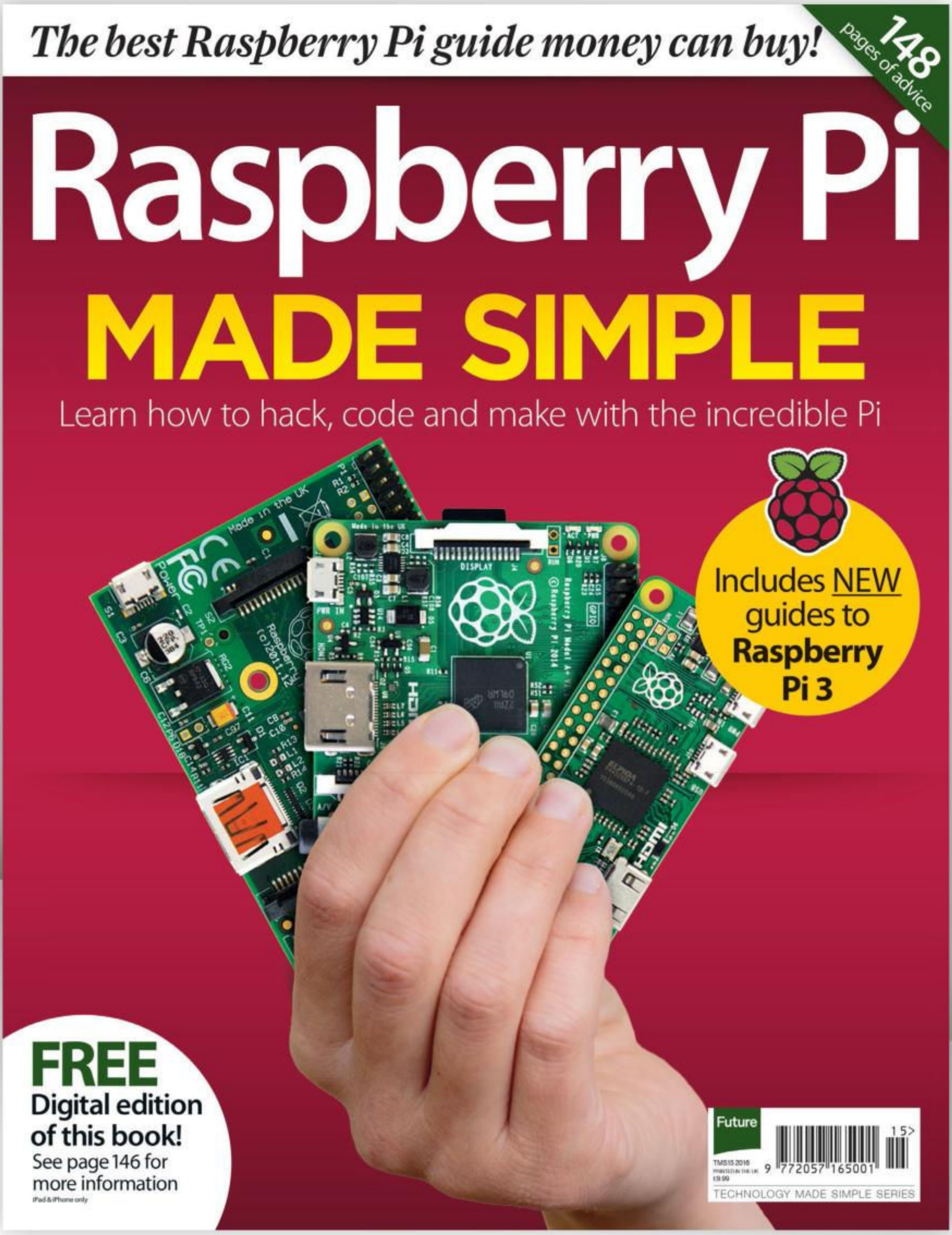
This page should appear if lighttpd is installed correctly. Feel free to replace it if you wish – it exists only to confirm the server is up and running.

If you're worried about your images being stolen and reused, the **index.php** file can be configured to include a watermark for images. See the readme file for more details.

Should you run into further problems, visit the developer's website at **http://sye.dk/sfpg** and follow the steps there to download the SFPG test script. This will output messages that can be helpful for troubleshooting. ■

GET THE MOST FROM
YOUR RASPBERRY Pi

OUT NOW!
WITH
FREE
DIGITAL
EDITION


The best Raspberry Pi guide money can buy!
148 pages of advice
Raspberry Pi
MADE SIMPLE
Learn how to hack, code and make with the incredible Pi
Includes NEW guides to **Raspberry Pi 3**
FREE
Digital edition of this book!
See page 146 for more information
iPad & iPhone only
Future
TECHNOLOGY MADE SIMPLE SERIES

Future

ORDER YOUR COPY TODAY!
Order online at **www.myfavouritemagazines.com**
Also available in all good newsagents

Available on the App Store
GET THE MAGAZINE ON Google play

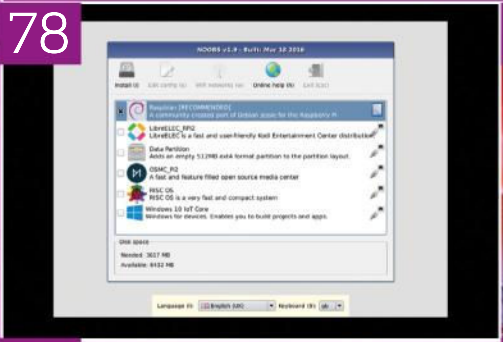
GET STARTED

Take your first steps with a Raspberry Pi...

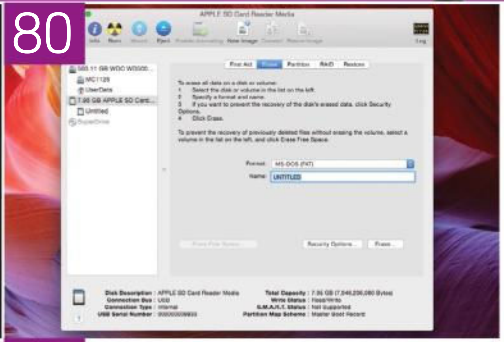
76

NEW TO THE RASPBERRY PI? OUR QUICK-START GUIDE WILL GET YOU UP AND RUNNING

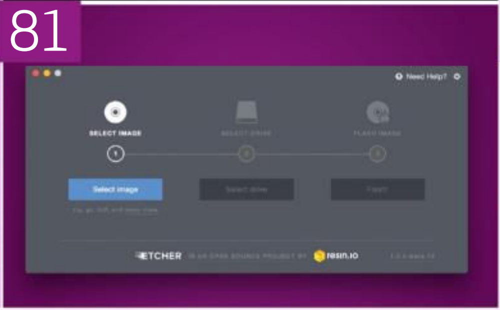
78


Install (i)
Online help (h)
LibreELEC_RPi2
Data Partition
OSMC_Pi2
RISC OS
RISC OS is a very fast and compact system
Windows 10 IoT Core
Windows for devices. Enables you to build projects and apps.

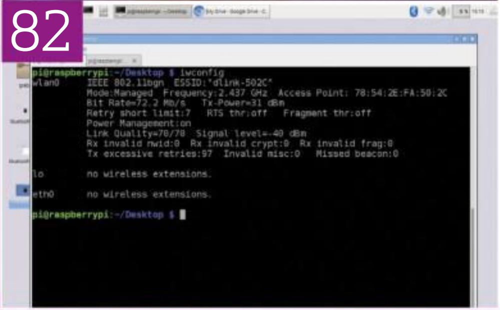
80



81


Need Help?
SELECT IMAGE
Select image
ETCHER

82



```
pi@raspberrypi:~/Desktop $ iwconfig
wlan0     IEEE 802.11bgn  ESSID:"dlink-502C"
          Mode:Managed  Frequency:2.437 GHz  Access Point: 78:54:2E:FA:50:2C
          Bit Rate=72.2 Mb/s   Tx-Power=31 dBm
          Retry short limit:7   RTS thr:off   Fragment thr:off
          Power Management:on
          Link Quality=70/70  Signal level=-40 dBm
          Rx invalid nwid:0  Rx invalid crypt:0  Rx invalid frag:0
          Tx excessive retries:97  Invalid misc:0   Missed beacon:0

lo        no wireless extensions.

eth0      no wireless extensions.

pi@raspberrypi:~/Desktop $ █
```

76 Get started with a Raspberry Pi
78 How to install Raspbian
80 Install NOOBS with any OS
81 Use Linux to install Raspbian
82 Using networks and the internet

Get started with a Raspberry Pi

Before you can actually do anything with your Raspberry Pi, you're going to need to hook up all the right peripherals. Here's how...

Even though it might not look like it, the Raspberry Pi is in a fundamental way just like any other desktop computer you might have owned or used in the past. That is to say, it has a processor, memory and storage and, in order for you to use it, it requires being connected to all the usual peripherals – a display, inputs and more. For most people, this should be straightforward, but there are some finer points that you should know about – details about the correct SD card to use and compatible wireless adaptors, for example – before we can move on to creating a boot disc to start it up.

To use the Pi as a computer, you'll need to connect at least a keyboard, a mouse, a display and a compatible SD card. In today's environment, it's going to be useful to at least have a wired internet connection around, or a suitable USB wireless adaptor for wireless networking. The final touch would be desktop speakers. On a technical level, peripheral support is provided by the Linux kernel that powers the Raspbian OS, but more on that later...

Need input!

Ever since the Model B+, the Raspberry Pi has been well furnished with four USB ports. This ensures that it's easy to connect all the peripherals you need. The base minimum of these is a USB mouse and keyboard for standard input – especially considering you're able to get keyboards that provide additional USB ports, helping to save one or more on the Raspberry Pi itself.

Pretty much any keyboard and mouse should work, but don't expect support for any unusual buttons (you'd be surprised at how many do work, though...).

The important thing to remember is that even with just four USB ports – or the single port on the Pi Zero – it's easy to expand your USB connectivity with a hub. For devices that don't require additional power, an unpowered hub is an option, while powered hubs can handle any devices, no matter whether power is required or not.

Display me

Technically, the Raspberry Pi works with most types of displays using the correct cable or adaptor. It's envisioned to be used with an HDMI monitor but works with older monitors with a DVI adaptor or an HD TV. With the HDMI connector, both the Pi and the display should automatically configure themselves correctly; we've successfully used the Raspberry Pi on a 4K display without issue, beyond manually setting the higher 4K screen resolution.

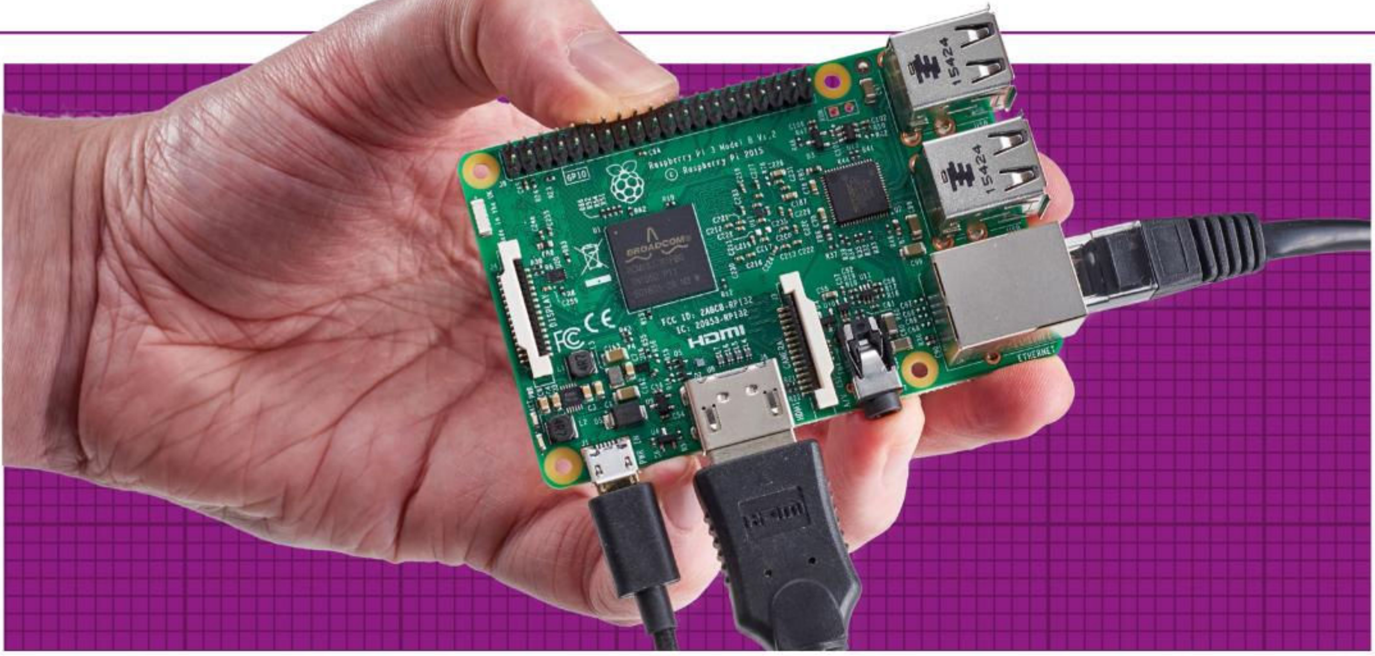
It's possible to connect the Pi to an old TV or monitor using an analogue composite connector – this is usually a round yellow RCA connector. With the Pi, the 3.5mm jack that's usually used for stereo audio output carries an extra line for the video. To access this, you need a 3.5mm jack to stereo and composite RCA.

Sounds good

The Raspberry Pi is perfectly capable of producing plenty of audio, and later we'll see projects that enable the use of the Pi as an audio streamer and media centre. You can get audio out of standard Pis in two ways. The first is over the HDMI line. When you plug the Raspberry Pi into a display that also has speakers, audio generated by the Pi is automatically sent over the HDMI line, too.

ON THE CASE

We admit, it can be unnerving having a naked Raspberry Pi sitting on your desktop, and it's certainly true that it could short on random wires, screws or metal surfaces. So, a common extra that people get for their Pi is a suitable case. At this point, there's quite a range available, and many leave the GPIO pins and other ports open for easy access. There are more exotic options that can mount the Pi within a display, a desktop-style case that expands its capabilities, and even a laptop chassis, so you can take your Pi with you. Just be aware that this means you can spend anything from £5 up to £100 or even more for your Pi case.



The alternative option is to use the standard 3.5mm audio jack, which can easily be plugged into most speakers.

Network and power

We're going to look into Pi networking in more detail over the page, largely because we want to look in full at wireless networking adaptors and the Pi 3's built-in wireless capabilities, alongside the Bluetooth features and how they work. Besides this, all Model B Pis come with a wired Ethernet network port. This can be plugged into any router, and boom – instant network access, including internet access if that router is online. For the bare-bones Model A Pis and Pi Zero, a USB wireless adaptor is required.

All Raspberry Pis use a micro USB port to supply their power. This has the advantage that, on the whole, you can power a Pi from almost any phone power adaptor. The power requirements vary, depending on which Pi you have and what you're doing with it. Old power supplies that provide only 5W should cope with a basic Pi Model B setup, which draws 3.5W, but the more peripherals you add, the more power it requires. The Pi 2 and 3 use 4W, so the draw is more critical. Ideally, you want a 10W (2A, 5V) supply, which is more common these days because modern tablets and phones also have higher power requirements.

At this point, you would have your Raspberry Pi all set to be a normal desktop PC. The fact is, the Pi was envisioned to be that and so much more. A big part of its extra abilities comes through the bank of GPIO (General Purpose Input Output) pins. We'll explore these later through fun and exciting projects, but for now, it's enough to know that these enable the Pi to control, monitor and power external devices and projects. You don't have to use them, but they're always there if you want them. Additionally, the Pi comes with a camera interface – although you can also use standard USB cameras – plus there's a dedicated digital touch display interface, too.

All you have to do now is follow our installation guide over the page and boot it up!

Connect it up and a Raspberry Pi will form the heart of a powerful but very compact computer setup.

SD CARDS

The main storage for a Raspberry Pi is an SD card, typically a microSD card, though the original Model B and Model A Pis used a full-sized SD card. On the whole, you can just run out and buy any microSD card and it should work, but it certainly makes sense to ensure that the card you buy will not only work, but will also be large enough and as fast as possible.

Not all SD cards are made the same. There are various speed categories, such as Class 6 (which denotes a minimum 6MBps write speed) and Class 10 (denoting 10MBps write speeds), along with the newer and faster UHS-1/2 (Ultra High Speed). One thing to keep in mind here is that the Pi's SD controller maxes out at 25MBps. It's also not capable of utilising the technology of the latest UHS high-speed SD cards.

With all that in mind, the key thing to look for in an SD card is to get one with fast read/write access, rather than pure throughout. It's also worth keeping an eye on the list of compatible tested SD cards at **http://elinux.org/RPi_SD_cards**.

You should also aim to get a card at least 8GB in capacity. The latest build of Raspbian – the main Pi OS – is now larger than 4GB. That said, given the price of SD cards, it makes sense to go for a 16GB or 32GB card. You need to balance the slight increase in price against the fact that a larger card is likely to be of use far longer.

SanDisk Ultra
32GB
micro SD HC I
10

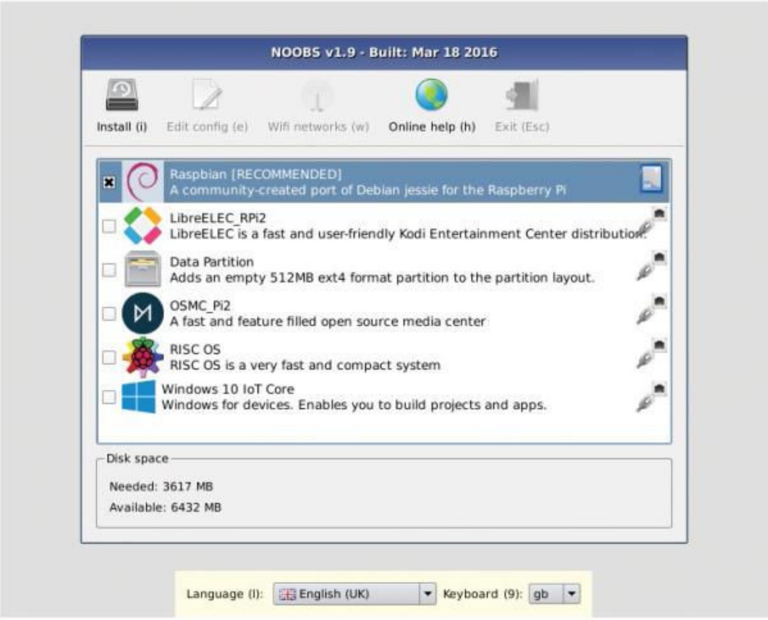
How to install Raspbian

Before you can do anything on your Raspberry Pi, you need some software for it to run. With a microSD card in hand, we guide you through the process

Before the Raspberry Pi can do anything (apart from sitting there looking cute) it needs an operating system (OS). The most popular OS for the Pi is Raspbian, which is based on Debian Linux. A number of other operating systems are also available for the Pi. Some of these can be downloaded from the Raspberry Pi Foundation website (**https://raspberrypi.org/downloads**) and some can be installed using NOOBS (over the page). There's Ubuntu MATE (Pi 2 and 3 only), an ARM port of Arch Linux, the classic RISC OS, the OSMC media centre, not to mention the RetroPie vintage gaming platform. There's also Snappy Ubuntu Core and even Windows 10 IoT Core. If you're a beginner, Raspbian is good to start with.

Unlike traditional computers, the Pi has no internal storage, so the OS needs to be loaded from an SD card. The original Pi models (models A and B) use full-size SD cards, whereas newer models (B+, 2, 3 and Zero) use the smaller (and much easier to lose!) microSD variety. SD cards pre-loaded with an OS can be purchased from pretty much wherever the Pi is available, and are commonly included in bundles, but it is also straightforward (and cheap) to make your own. This can be done using freely available tools on Linux, macOS or Windows.

If you already have a pre-loaded SD card, you can skip this section entirely. If not, the most straightforward way to get going with the Pi is to download the latest Raspbian image and write it directly to an SD card, which needs to be at least 4GB in capacity. If all this seems too technical, you may wish to look at NOOBS (over the page). Note that on macOS computers, it's also possible to write the Raspbian image using the **dd** tool from the command line. If you're comfortable working this way, the procedure is very similar to the Linux instructions on page 81. Alternatively, check out the instructions on the Raspberry Pi Foundation website at **www.raspberrypi.org/documentation/installation/installing-images/mac.md**.



☒ Raspbian [RECOMMENDED]
A community-created port of Debian jessie for the Raspberry Pi

☐ LibreELEC_RPi2
LibreELEC is a fast and user-friendly Kodi Entertainment Center distribution.

☐ Data Partition
Adds an empty 512MB ext4 format partition to the partition layout.

☐ OSMC_Pi2
A fast and feature filled open source media center

☐ RISC OS
RISC OS is a very fast and compact system

☐ Windows 10 IoT Core
Windows for devices. Enables you to build projects and apps.

Disk space
Needed: 3617 MB
Available: 6432 MB

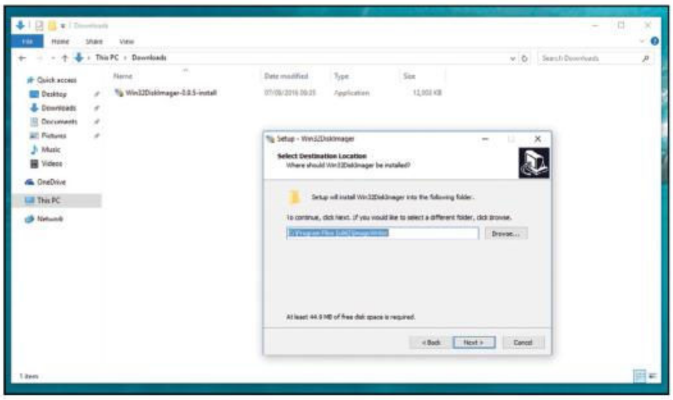
Language (l): English (UK)
Keyboard (9): gb

› Raspbian should be your first choice, but the other OSes can be fetched and installed with minimum fuss.

Enjoy Raspbian

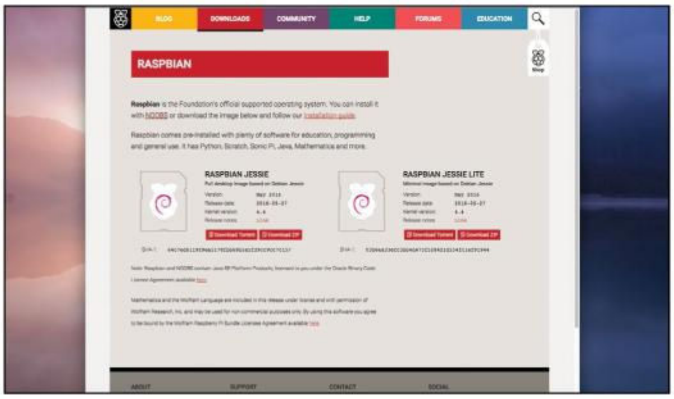
Once your SD card is ready (whichever road you choose), you're all set to boot up your Pi. Remove the SD card from your computer, plug it into your Pi (along with all the other gubbins – display, power, keyboard and mouse), power up and it should boot up to the raspi-config program. From here, you should expand the filesystem (if your card is greater than 4GB), change the default password and enable Boot to Desktop. Select Finish and you'll be able to reboot to the Raspbian desktop. There's all manner of interesting things you can play with right away, including Minecraft, Wolfram Alpha and Sonic Pi. If you have a wired network connection, then that will work out-of-the-box. Some wireless ones will as well, but for many, this will be their first encounter with the recalcitrance of Linux. Just be patient, and remember, Google search is your friend…

INSTALL RASPBIAN WITH WINDOWS OR MACOS



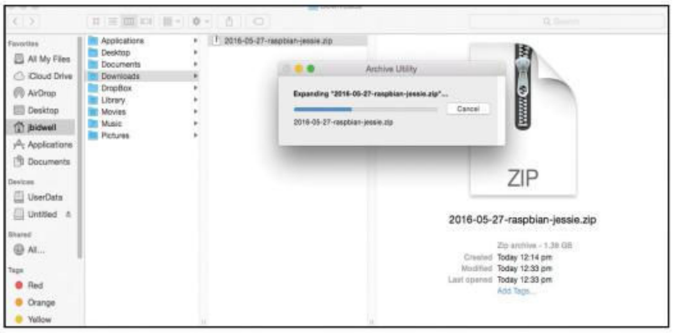
1 SD card tools

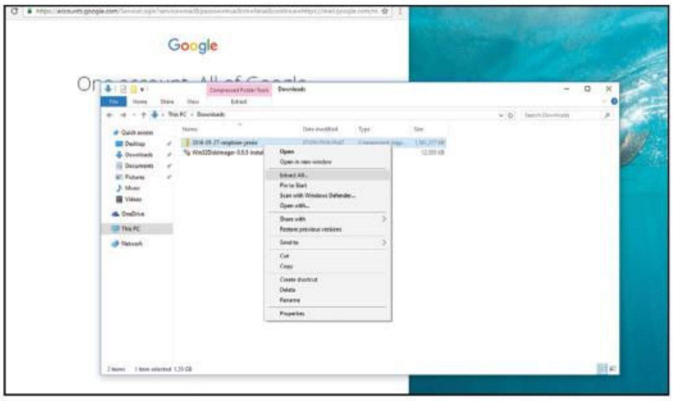
We first have to download some additional image-writing software. For Macs, the simplest tool is Rpi-sd Card Builder (**http://alltheware.wordpress.com/2012/12/11/easiest-way-sd-card-setup**), but you may also want to look at Pi Filler (**http://ivanx.com/raspberrypi**) or ApplePi-Baker (**www.tweaking4all.com/hardware/raspberry-pi/macosx-apple-pi-baker**). On Windows setups, we recommend using Win32 DiskImager, which can be downloaded from **https://sourceforge.net/projects/win32diskimager**. Installation of all of these programs is straightforward.



2 Download Raspbian image

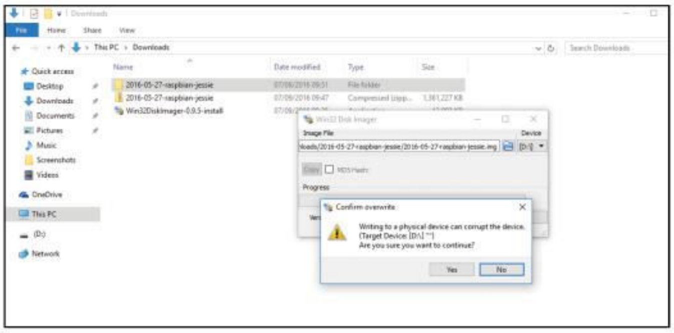
You'll find the Raspbian image at **www.raspberrypi.org/downloads/raspbian**. The current edition is called Jessie, the same as the Debian release it's based on. Be sure to choose the full edition, rather than the Lite one, at this stage – unless, of course, you're sure that you're happy at the command line, with no desktop environment. You may wish to save the Foundation some bandwidth and use the BitTorrent link. This requires you to have a BitTorrent client installed, such as Transmission or uTorrent. Whatever your download method, the full Raspbian image is a 1.3GB download, so will take a few minutes.

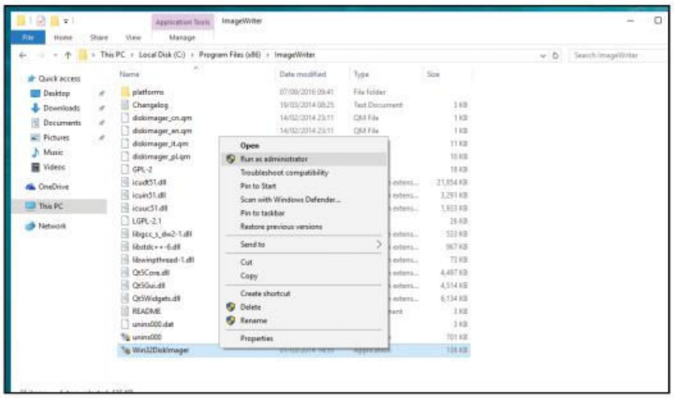




3 Unzip file

You'll end up with a file in your downloads folder, called something such as **2017-01-11-raspbian-jessie.zip**. Make sure that you're not running short of hard disk space, because this file will be close to 4GB in size when it's uncompressed. Double-click the Zip file (if you're using a Mac), or right-click it and select Extract All (on Windows). It should extract to a file called something along the lines of **2017-01-11-raspbian-jessie.img** in the same folder as the downloaded Zip file. This is the image file that we need write to the SD card in the next step.





4 Write the image

Win32 Disk Imager needs special privileges to write to the SD card. So, instead of double-clicking its .exe, right-click it and select Run as Administrator. Now select the image file you extracted in the previous step. In the Device section, pick the drive letter of the SD card. Click Write, double-check you have the right device, and writing begins. Wait for it to complete and, hey presto, all done. Mac tools work similarly (you may be asked for a root password), but be sure to select the correct SD card device, not your hard disk. Whatever destination you choose is wiped, so you don't want to get this wrong.

Install NOOBS with any OS

Possibly the easiest way to get a new OS up and running on your Pi can be achieved using any operating system – as long as you know how!

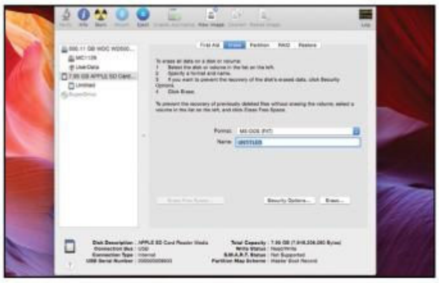
NOOBS (New Out Of the Box Software) makes it easy to install Raspbian (or other OSes) on your Raspberry Pi. You can buy an SD card with NOOBS on it from many vendors, or we'll show you how to do it yourself below.

With a correctly-prepared NOOBS SD card, the Pi can boot to a menu offering you a choice of OSes. Pick one and NOOBS installs the OS on the SD card, so it will load the next time the Pi is rebooted. Or you can hold down Shift as the Pi boots to use the NOOBS installer once again.

Preparing a NOOBS SD card with Raspbian on it requires at least a 4GB SD card – other OSes (in particular, Windows 10 IoT Core) will require 8GB. The simple process of installing NOOBS to an SD card is outlined below. Note that if you've bought an SD card with NOOBS already on it, you don't need this section – just plug in and turn on.

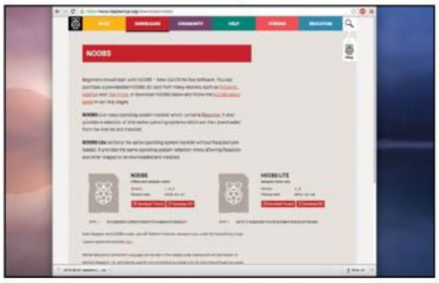
Once your card is ready, reboot the Pi with power, mouse, keyboard and monitor all connected. You should see a list of OSes (tailored for your Pi model). The Raspbian entry should have a picture of an SD card on the right, indicating that it can be installed even if the Pi is not connected to the internet. Tick the box to its left, hit Install and, before long, you'll be able to reboot straight into Raspbian.

INSTALL RASPBIAN WITH WINDOWS OR MACOS



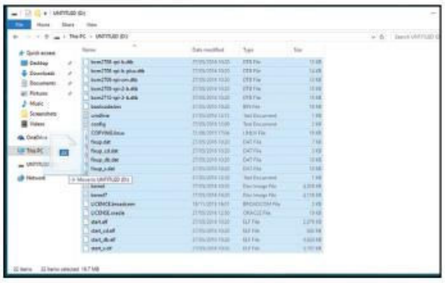
1 Format the SD card

Before you can do anything, your SD card must be correctly formatted. Freshly purchased blank cards tend to already be formatted as FAT32, but if you're recycling an old one, you need to do this yourself. First make sure there isn't anything important on it – it will be erased. Formatting can be done directly from Windows Explorer (right-click the SD card and choose Format) or from Disk Utility in macOS (go to the Erase tab). In both cases, choose FAT32, not the NTFS or HFS+ filesystems. Linux users can format SD cards using Gparted, Gnome Disks or the command line with `mkdosfs -F32 /dev/sdX1`, for example. Note that NOOBS requires that only a single FAT32 partition exists on the card, so if there are others, these should be removed with appropriate partitioning tools.



2 Download NOOBS

NOOBS is available in two forms: the regular edition (weighing in at 1GB, zipped) and NOOBS Lite (a mere 27MB). The Lite edition does not include the Raspbian image – it has to be downloaded from within the NOOBS installer. The full-fat edition allows Raspbian to be installed without an internet connection, but other OSes have to be downloaded. Again, this is easily achieved from the comfort of the NOOBS installer. Both versions can be downloaded from **www.raspberrypi.org/downloads/noobs**, either directly or via BitTorrent. The latter is a little more considerate of bandwidth but requires a BitTorrent client. Many Linux distros include the Transmission BitTorrent client (also available for macOS), and there are plenty to choose from on Windows (uTorrent, Deluge, Bitcomet...).



3 Unzip and copy

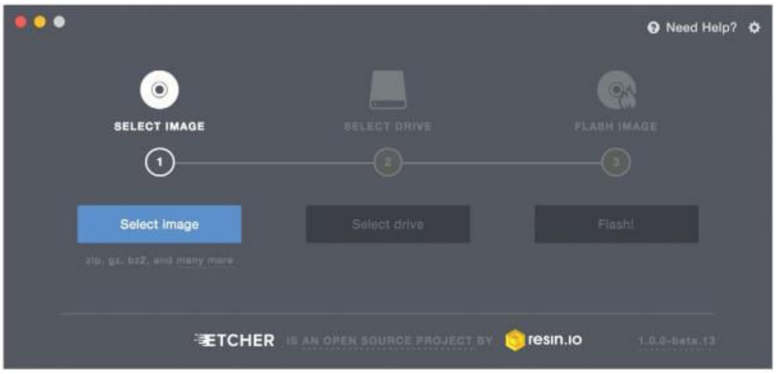
You should have a file named something like **NOOBS-V-2-1-0.zip** in your downloads folder. This must be decompressed and the contents put on the SD card. On Windows and Mac, you should be able to extract the files by just double-clicking the Zip file. The files extract to a folder named something like **NOOBS-v2-1-0**. Now they must be copied to the SD card. Be sure to transfer only the files inside the NOOBS folder, rather than the folder itself, otherwise it won't work. Open the **NOOBS-v2-1-0** folder and select all the files (click the first one, hold Shift, then click the last one). Drag them on to the SD card icon and wait for them to finish copying (the full version can take a while). Right-click and safely eject the SD card. You should now be able to use this card to boot your Pi, as explained above.

Install Raspbian with Linux

If you're running any flavour of Linux on your computer, then you have an even easier life when it comes to getting Raspbian on to your Pi

Linux users can get Raspbian set up from the comfort of their operating system, too. Much the same as on Windows and Mac, the first steps are to download the Raspbian image (either from a web browser or using Wget) and unzip it. There are a few different options for writing the image. You can use a graphical tool such as Etcher (see **www.etcher.io**) or Ubuntu's Startup Disc Creator. These work very much like the image-writing tools for macOS and Windows. As with most tasks on Linux, the usual approach is to use the command line.

Insert the target SD card into the machine. We'll use the **dd** tool, which is part of all standard Linux installations, to transfer the image. You definitely don't want to accidentally wipe your hard drive this way (and it's possible to do just that if you get this wrong), so first use the `lsblk` command to see which device node it is attached to. You should be able to identify the SD card by its capacity. It could be **/dev/sdb** or it could be **/dev/sdc**. Newer PCs that have a native MMC controller will give it a slightly different name, such as **/dev/mmcblk0p1**. Note that it is the device, not partitions on that device (such as **/dev/sdb1** or **/dev/mmcblk0p1n2**), that we are interested in. Any existing partition information will be lost when we transfer the image to the device. Writing the image to a partition won't work at all well, so don't do that.



The *Etcher* image writing program is probably the slickest way to make SD cards under Linux, and saves you from getting your hands dirty at the command line.

We'll refer to the SD card as /dev/sdX (replace the X with the actual drive letter) and assume you've unzipped the Raspbian IMG file to the Downloads directory. In a terminal type:

```
$ sudo dd if=~/Downloads/26-05-16-raspbian.img of=/dev/sdX bs=1M
```

This process can take a long time, particularly on older or cheaper SD cards. Sometimes, even when the command completes (when you are able to enter another command), the job is not done. If your SD card or adaptor has an activity light, then you can see if anything's still being transferred, but if not, try the `sync` command. If it doesn't complete right away, data is still in-flight. Once everything is ready, you can remove the SD card, boot your Pi and tweak the raspi-config program as per the Windows and macOS instructions.

Linux luck

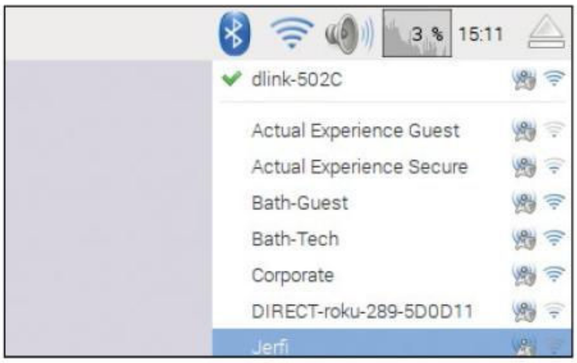
Under Linux, there is a slight advantage in that the SD card can be browsed natively. It's only a slight advantage because things can be sent over the network to side-step filesystem incompatibilities amongst OSes. Still, it can sometimes be handy (for example, if the Pi is shut down) to just be able to grab the SD card, slip it in a slot, and grab photos or edit configuration files.

If you're not (yet) a Linux user, it's also possible to use the live disc mode of a distribution (Ubuntu, Fedora, openSUSE, Mint, for example) and create the SD card from there. There's a slight problem downloading the image to the root filesystem, however, because this is all in RAM for live discs (their modus operandi is to not touch your hard drive), and there typically isn't enough space for several gigabytes of OS image. This can be worked around, though, because the image can be downloaded to a USB stick beforehand and copied from there. On Ubuntu, USB sticks are mounted in a directory inside **/run/media/ubuntu**, so change the `if` parameter of the `dd` command accordingly. Alternatively, you can use the Startup Disk Creator on an Ubuntu live DVD if you don't feel like typing things. ■

Using networks on the Pi

Getting hooked up to a local network and the wider internet is easier than you might think, as long as you have all the right bits...

Everyone (and, it seems, everything) is online these days, and that means that if you want your Raspberry Pi to do anything "interesting" – even if that's just browsing Facebook – you need to get it networked. As with everything Raspberry Pi, it's made as hard as possible, to ensure you learn as much as possible along the way! No, actually, the only complication is that Raspbian doesn't hide any of the nitty gritty of networking from you, combined with the need to add wireless dongles to all Raspberry Pi models except the latest Pi 3, which comes with built-in wireless and Bluetooth.



Raspbian now supports easy GUI-based wireless connections – so modern!

The first thing you need to do before buying a wireless adaptor is to check whether the one you already have or plan to get will work with your Raspberry Pi. That's where **http://elinux.org/RPi_USB_Wi-Fi_Adapters** comes in – this is a huge list of both verified adaptors and those that cause issues.

The good news is that a huge range works with Raspbian, thanks to its Linux kernel. It comes with built-in support for a large number of adaptors. Because of the way Raspbian is built, an adaptor that doesn't work with Raspbian might work with another Linux OS, such as Arch or OpenELEC, because of the particular version of the Linux kernel they ship with.

You can confirm whether a USB device is working correctly – other than the Raspbian wireless being enabled – by opening a terminal and typing `lsusb`, then pressing Return. This provides a list of all your USB devices, and one should be something that sounds like your USB Wi-Fi device.

A second command, `iwconfig` provides a list of working wireless adaptors connected to your Pi. This offers a slightly more direct confirmation that the USB dongle is connected and working.

An easy way to avoid any worry is to buy an official Raspberry Pi Foundation Wi-Fi dongle. Designed to work with all Raspberry Pis and Raspbian, this eliminates any worries. There's also the added point that it helps fund the Foundation, too. You can get more details from **www.raspberrypi.org/products/usb-wifi-dongle**.

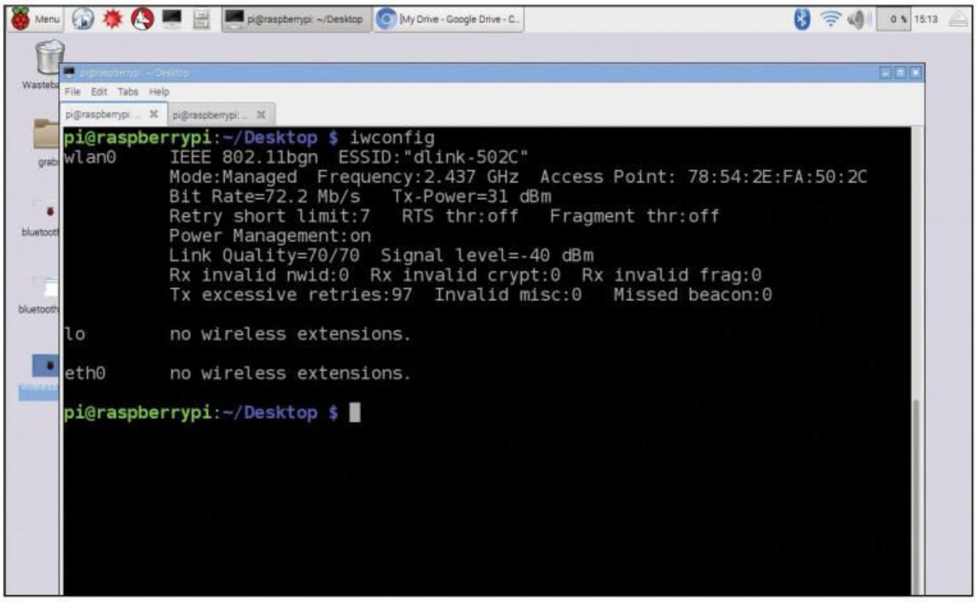
Back in the day, you had to edit text files to configure your wireless dongle, but these days Raspbian has been fully updated to provide a GUI (or Graphical User Interface) for connecting to any wireless network. How civilised! Wi-Fi connections can be made via the familiar network icon at the right-hand end of the menu bar. If a working Wi-Fi dongle is plugged in, left-clicking this icon brings up a list of available Wi-Fi networks, as shown in the picture on this page. If no networks are found, it shows the message "No APs found – scanning..." Wait a few seconds more without closing

TCP AND YOUR SORE IP

If you're new to PCs and networking, then the sudden talk of IP numbers and the like might be confusing. Let's just say that local networks and the internet all use a very clever system called TCP/IP. On a pedantic level, that's two – yep, two – communications protocols: TCP, alias Transmission Control Protocol; and the IP part, alias Internet Protocol. You don't really need to know anything about it, other than this: it assigns a unique number to each and every device on the internet. This enables messages to get to and from your devices, finding their way to the right place.

Each device gets an address made up of four numbers, separated by full stops. Each number can be between 0 and 255. So, typically, an address would be something like **192.168.0.1** or **192.168.100.23**. Generally, your home and business network has its own IP range – the broadband router you use to connect to the internet creates this automatically using something called a DHCP server, and uses Network Address Translation, or NAT, to route messages to and from your local network and the wider internet. See, it does get awfully complicated very quickly!

Usually, when we talk about IP addresses, it's when you want to know the address of a device or server on your network or on the internet, or you want to give a device a specific address. Luckily, you won't have to know much more than this!



```
pi@raspberrypi:~/Desktop $ iwconfig
wlan0     IEEE 802.11bgn  ESSID:"dlink-502C"
          Mode:Managed  Frequency:2.437 GHz  Access Point: 78:54:2E:FA:50:2C
          Bit Rate=72.2 Mb/s   Tx-Power=31 dBm
          Retry short limit:7   RTS thr:off   Fragment thr:off
          Power Management:on
          Link Quality=70/70  Signal level=-40 dBm
          Rx invalid nwid:0  Rx invalid crypt:0  Rx invalid frag:0
          Tx excessive retries:97  Invalid misc:0   Missed beacon:0

lo        no wireless extensions.

eth0      no wireless extensions.

pi@raspberrypi:~/Desktop $ 
```

The iwconfig command helps check whether a wireless adaptor is working.

the menu, and it should find your network.

The icons on the right show whether a network is secured or not, and its signal strength. Click the network that you want to connect to; if it is secured, a dialog box appears, prompting you to enter the network key. Do so and wait a couple of seconds. The network icon flashes briefly to show that a connection is being made; once it is ready, the icon stops flashing and shows the signal strength.

Older versions of Raspbian provided a Wi-Fi Config tool on the desktop. There's no reason why you should encounter an older version at this point, but if you do, it's no more complicated than running it and then using its Scan button to locate your desired wireless network.

Wired networks

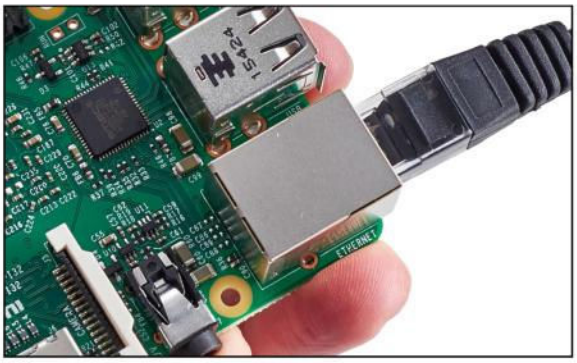
We should also mention wired networks while we're here. All Model B Raspberry Pis have a wired network connection called an Ethernet port. This uses an Ethernet cable to connect to a standard router. A wired connection remains the most reliable and the fastest option for transferring files and getting an internet connection. In terms of having to do anything, you don't – just plug in the cable, and as long as it's connected to a working router that already has an internet connection, Raspbian is allocated an IP address.

Most Ethernet ports provide two status LEDs (the original Pi did not; these status lights were moved to the board itself). The green confirms a working connection (any blinking shows activity), while the amber one announces a full-speed 100Mbps connection (10Mbps if it's off).

It's also worth noting that the Linux kernel can happily handle multiple network connections over different network adaptors. This means you can connect to a different network or the same one via both a wired connection and wireless at the same time – the kernel's TCP library happily balances transfers over both the adaptors. It also means you can remotely connect to the Pi via either assigned IP address.

As with USB Wi-Fi dongles, it's also possible to add wired Ethernet adaptors via a USB port – a full list of verified and problem adaptors can be found at **http://elinux.org/RPi_USB_Ethernet_adapters**. While this doesn't make total sense for a Model B board, it may be exactly what you need when it comes to the Model A or the Pi Zero.

The subject of networking can get highly involved, so we're going to leave things with a standard IP connection.

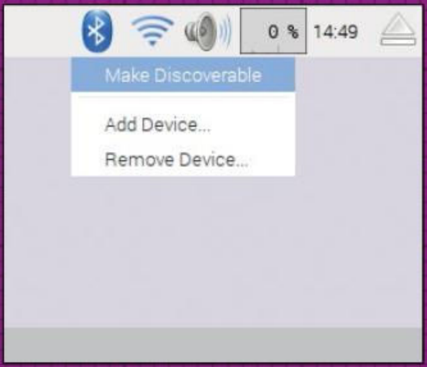


Wired networking is the fastest and most reliable way to connect things.

BLUETOOTH DEVICES

With the addition of the embedded dual wireless/Bluetooth abilities in the Pi 3, Raspbian was updated to support Bluetooth out of the box. Prior to this, you had to "hack" in the Bluetooth support, with the command `sudo apt-get install --no-install-recommends bluetooth` and, once it was installed, use `sudo service bluetooth status` to enable it.

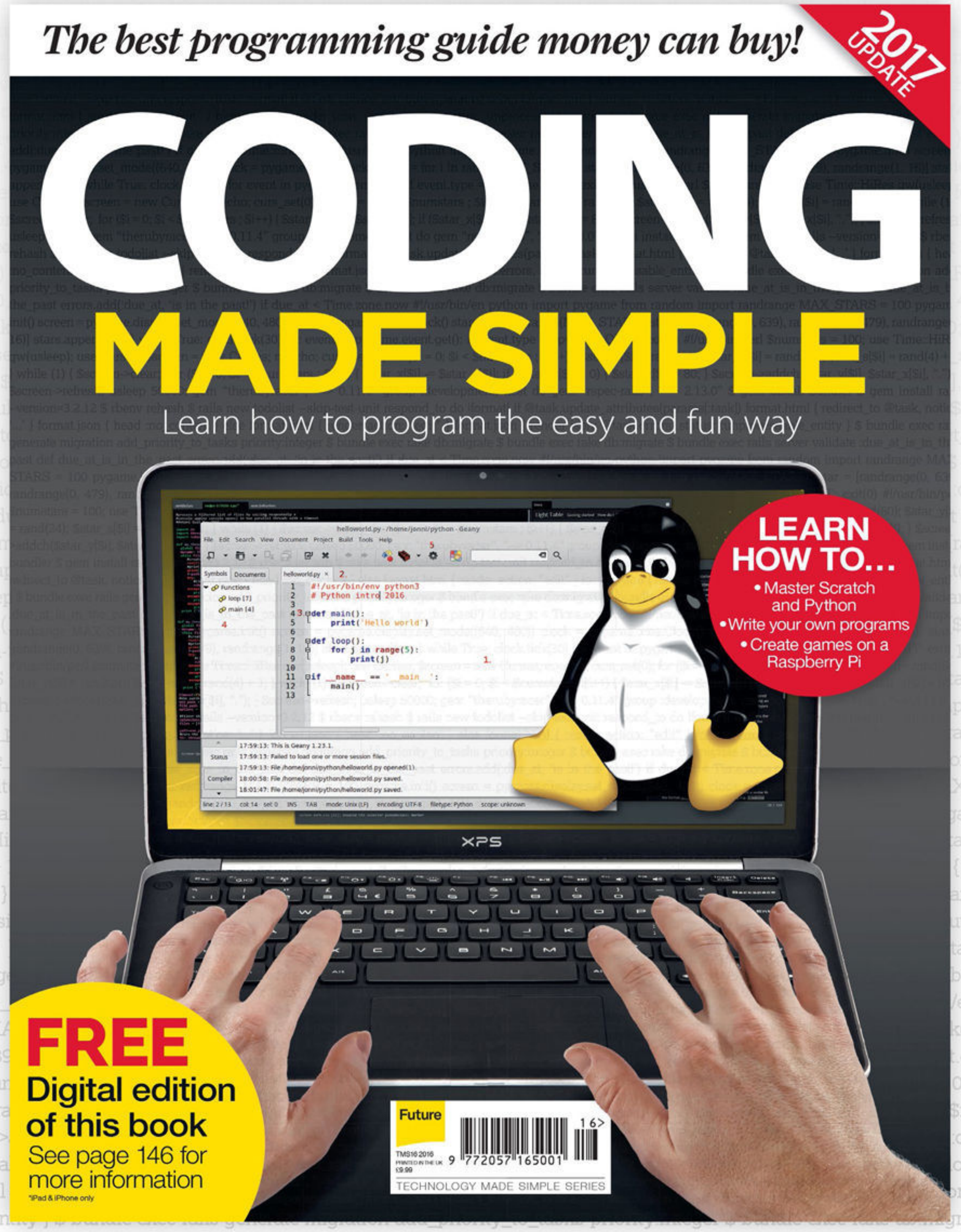
While this was a simple case of installing a manager tool, it's far neater having it around by default. If you use a smartphone you're likely to have used Bluetooth, with all the Bluetooth speakers, keyboards and headsets available these days, but if not, the good news it that it's far simpler than connecting to a wireless network. This goes for all Bluetooth devices, but you first need to make them "discoverable" – this enables other Bluetooth devices to then attempt to connect with them, at which point you have to allow those connections or not, as the case may be. Left-click the Bluetooth icon in the top-right of the Raspbian desktop, select Make Discoverable, then choose Add Device to pull up a list of nearby Bluetooth devices. Select the correct one and click Pair. With keyboards and devices, you often have to enter a code on the device to finalise the connection. You'll want to ensure that you choose Stop Discoverable afterwards.



The Raspberry Pi should pair with most Bluetooth devices without an issue.

EVERYTHING YOU NEED TO GET PROGRAMMING

OUT NOW!
WITH FREE
DIGITAL EDITION



The best programming guide money can buy!

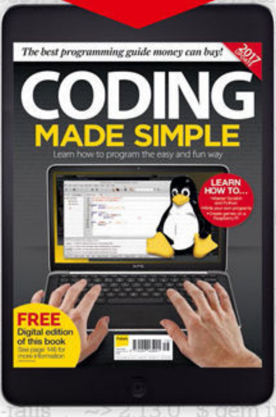
2017 UPDATE

CODING
MADE SIMPLE
Learn how to program the easy and fun way

LEARN HOW TO...
- Master Scratch and Python
- Write your own programs
- Create games on a Raspberry Pi

FREE
Digital edition of this book
See page 146 for more information
*iPad & iPhone only

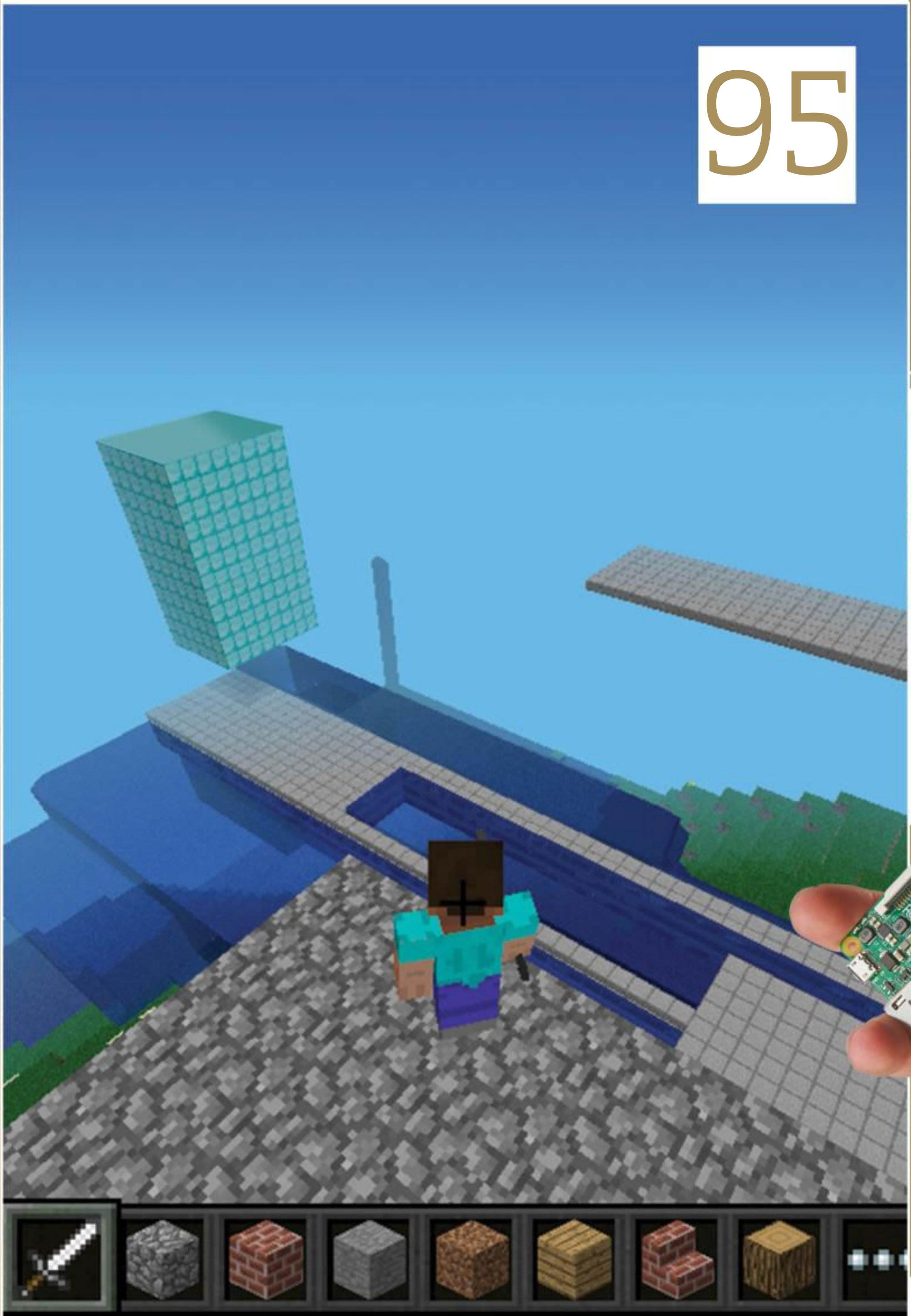
Future
TMS16 2016
PRINTED IN THE UK
£9.99
TECHNOLOGY MADE SIMPLE SERIES



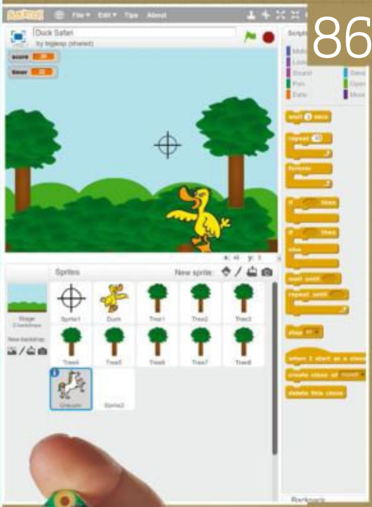
DELIVERED DIRECT TO YOUR DOOR
Order online at **https://www.myfavouritemagazines.com**
or find us in your nearest supermarket, newsagent or bookstore!

Pi CODE

Learn how to code in your language of choice



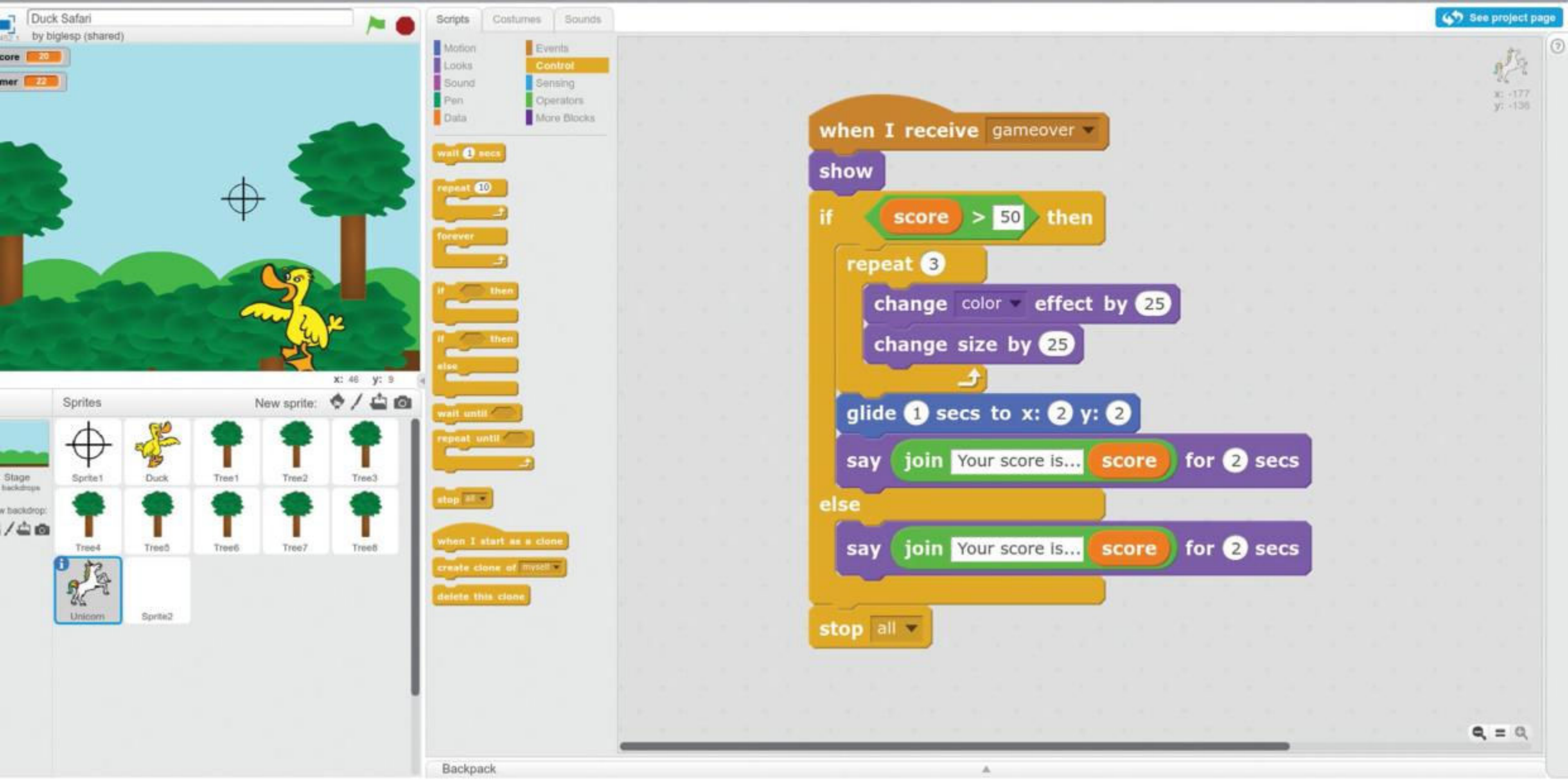
GIVE YOURSELF GODLIKE POWERS TO CREATE A SKYSCRAPER, A ROAD OR A RIVER... OR TO DESTROY



86 Design a game in Scratch 2

92 Get started with Python

95 Build your own Minecraft city



```
when I receive gameover
show
if score > 50 then
    repeat 3
        change color effect by 25
        change size by 25
    glide 1 secs to x: 2 y: 2
    say join Your score is... score for 2 secs
else
    say join Your score is... score for 2 secs
stop all
```

Design a game using Scratch 2

Les Pounder shows how you can create a simple console game using a few blocks of Scratch code and a creative imagination

The popularity of Scratch has soared since the Raspberry Pi was released. Children across the United Kingdom have started their coding journey using this effective tool. Why is it so popular? Scratch beautifully demonstrates the basics of Computer Science, loops, iteration, variables, arrays etc. But Scratch manages to keep all of these concepts relatively simple and easy for the user, typically a child, to use. When children first use Scratch, they simply treat it as a set of building blocks, connecting blocks together and seeing what they can create. But over time and with the help of the curriculum in our schools, children are learning key skills which can be applied to all languages, all thanks to Scratch.

The Raspbian operating system, the de facto standard for all versions of Raspberry Pi, comes with version 1.4 of Scratch, an older version of Scratch which has been tweaked to also work with the Pi's GPIO. But in late 2016 the Raspberry Pi Foundation introduced an update of Raspbian that now comes with the Pixel Desktop, the Chromium web browser and Flash – a contentious inclusion given that it is not free software, but required so that Scratch 2, the latest version, could be made available. The Foundation, always keen to ensure that they provide the relevant software for learners, chose to include Flash in order to ensure that users had access to the latest version of Scratch aligned to the latest learning resources.

In this tutorial we'll introduce a number of coding concepts, such as inputs, loops, variables and conditionals, by creating a clone of the Nintendo classic game Duck Hunt, which was popular in the 1980s on the NES console. So let's start our project with an overview of the new Scratch 2 interface.

Scratch 2 – take your projects anywhere

Scratch 2 is seen as a significant improvement over 1.4, largely thanks to being online. You don't need to sign up for an account, but the benefits of doing so are that you can share your work with others, and you can remix their work to create your own. You can also build your own blocks, which can contain extra functionality, much the same as a function in Python or a macro in Office applications.

Another big benefit of Scratch 2 is that you can access your projects from anywhere in the world using any computer. This enables children to work on their school projects at home, with no specialist software install, and it also ends "the dog ate my homework" scenarios. Children can share their projects in class blogs and similar channels by embedding their game in much the same way as we can embed YouTube videos into posts.

GETTING YOUR DUCKS IN A ROW

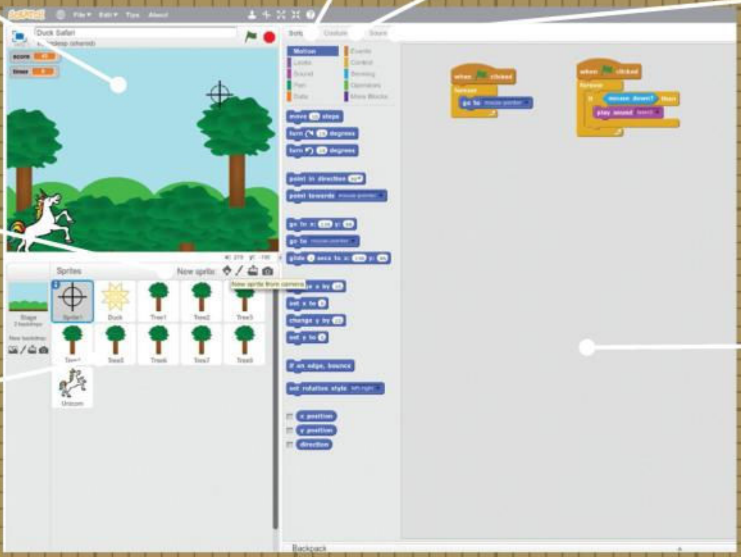
The Stage The Stage is where the user can interact with their project – for example sprites will move inside the stage, and players can control elements of their game in this panel.

Scripts All of the blocks that we use to code are contained in the Scripts tab. They are categorised by their function, and handily by colours.

Costumes Costumes are how we can change the look of a sprite. We can edit a sprite by hand or use the Sprite library to create a quick change.

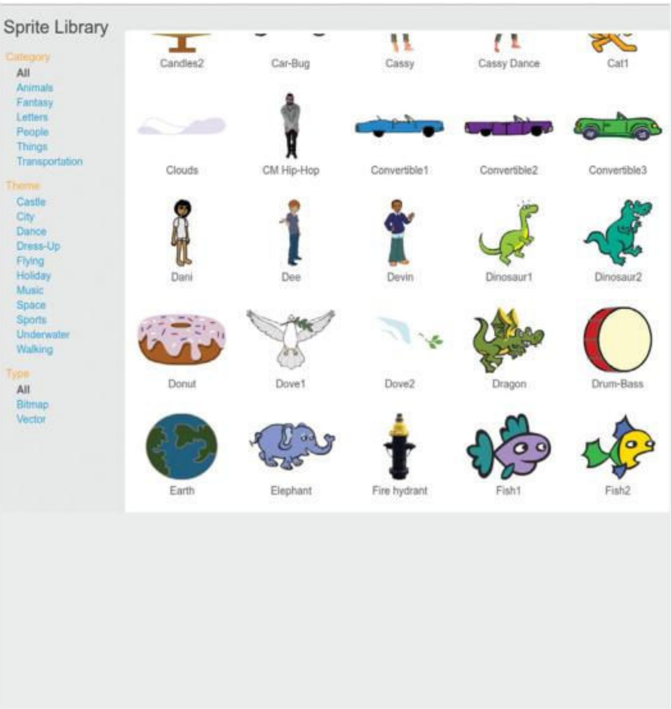
Sounds Sounds can be loaded via the library so that our sprites and the game as a whole can have sounds effects and music while we play.

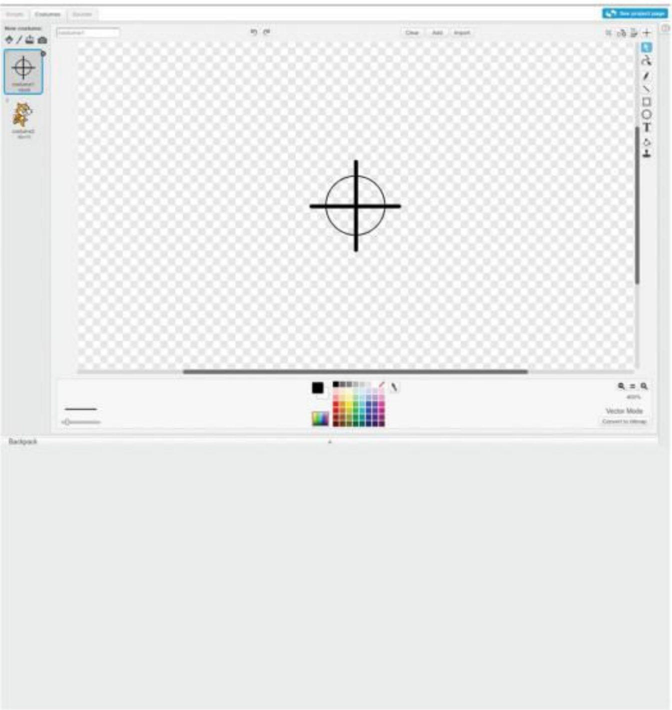
Sprite options Create a new sprite from a library, draw your own sprites, import from your computer, or use a webcam to take a picture to use.



Code The Coding area is where blocks can be placed and used to build up a project. The Coding area displays the code for the current sprite, so it is vital that you select the correct sprite before starting.

Sprites We can see all of the sprites for a project in the Sprites select area. Each sprite has its own blocks of code, and we can click on a sprite to start coding.


Sprite Library
Category
All
Animals
Fantasy
Letters
People
Things
Transportation
Theme
Castle
City
Dance
Dress-Up
Flying
Holiday
Music
Space
Sports
Underwater
Walking
Type
All
Bitmap
Vector
Candles2
Car-Bug
Cassy
Cassy Dance
Cat1
Clouds
CM Hip-Hop
Convertible1
Convertible2
Convertible3
Dani
Dee
Devin
Dinosaur1
Dinosaur2
Donut
Dove1
Dove2
Dragon
Drum-Bass
Earth
Elephant
Fire hydrant
Fish1
Fish2



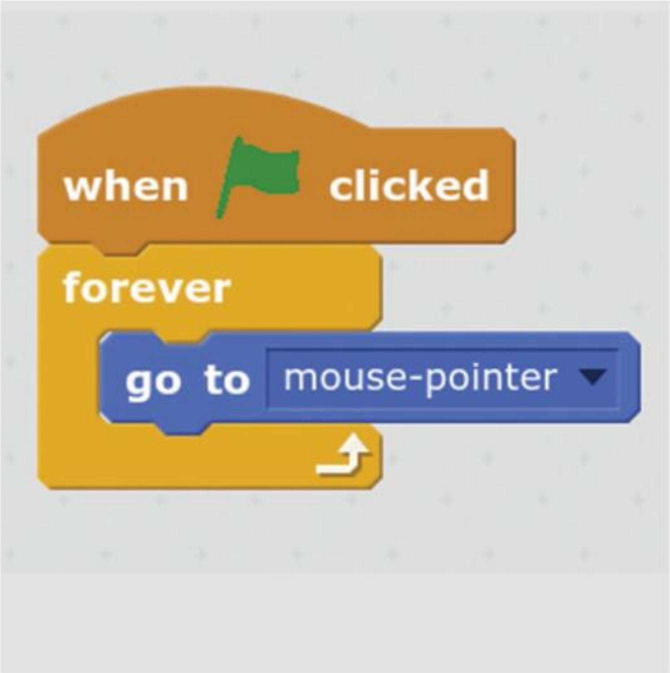
1 Creating our duck

We'll start the project by changing our default sprite, which is that iconic Scratch cat, into a duck. To do that we need to click on Costumes, then look for the first icon under New costume. In the costume library select the duck and click OK. We also added a star explosion costume for later. Delete the cat costume by clicking on the X in the top right of the sprite.

2 Creating our crosshairs

We now create a new sprite. Click on the paintbrush under the Stage to open the editor. The editor works in two modes, bitmap and vector. For this sprite, switch to vector and then draw your aiming crosshairs, ensuring that they are in the centre of the editor. When done, click the Scripts tab to go back to coding. Our crosshairs will be used to target the duck, ready for us to open fire.

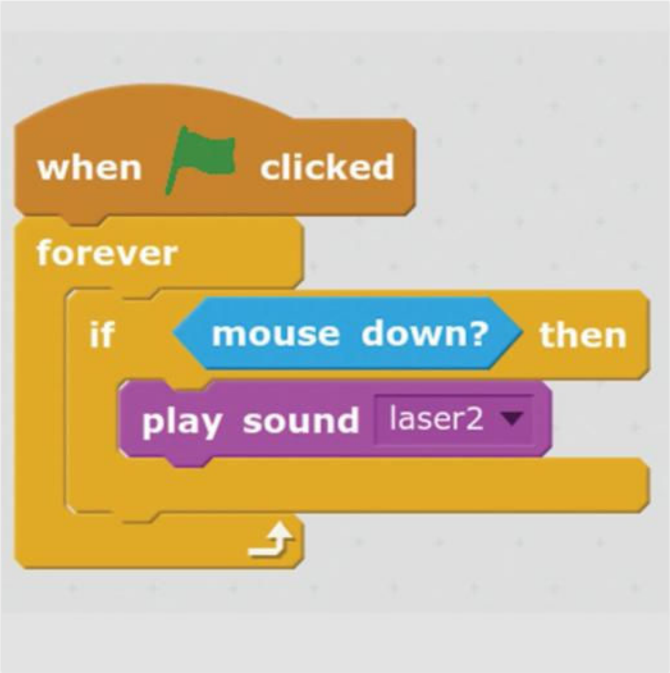
»



```
when [green flag] clicked
forever
    go to mouse-pointer
```

3 Coding our crosshairs

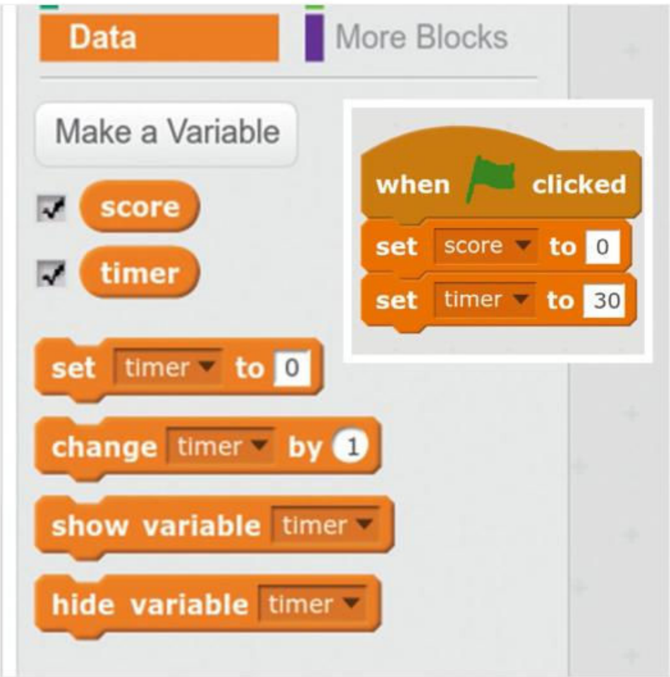
To enable an easy aim we code the crosshair sprite to follow our mouse pointer across the screen. First click Events in the Categories panel at top-left, and drag "When Green Flag clicked" into the Scripts panel. Next click Control and add a Forever loop. Now click Motion and add "Go to mouse-pointer" to the script. Now our sprite will follow our mouse, which is triggered by clicking on the Green Flag.



```
when [green flag] clicked
forever
    if mouse down? then
        play sound laser2
```

4 Adding sound

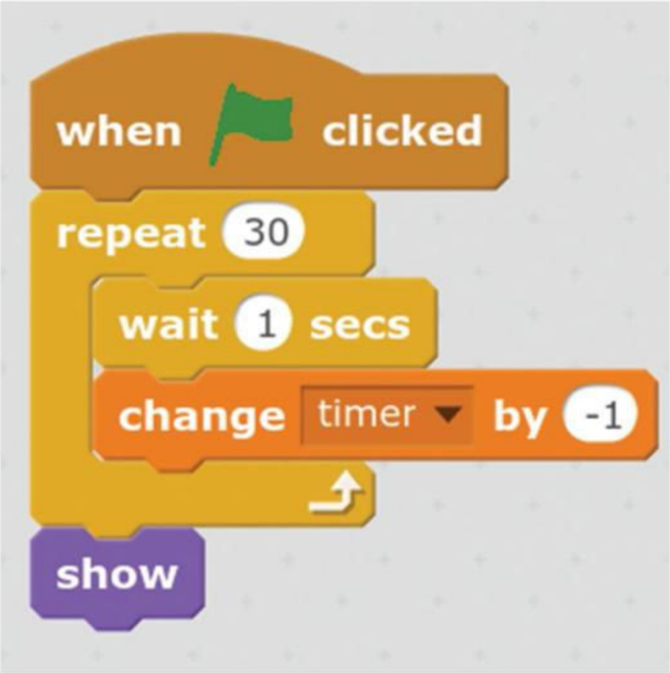
Adding sound is handled in the same way as adding sprites. Click on Sound, then click on the Speaker icon under New sound. From the library choose "laser2" and click OK. Now create the sequence of blocks shown, which will play a sound when the mouse button is pressed. Remember to place the blocks inside the Forever loop, so that Scratch constantly checks for a button press.



```
when [green flag] clicked
set score to 0
set timer to 30
```

5 Creating variables

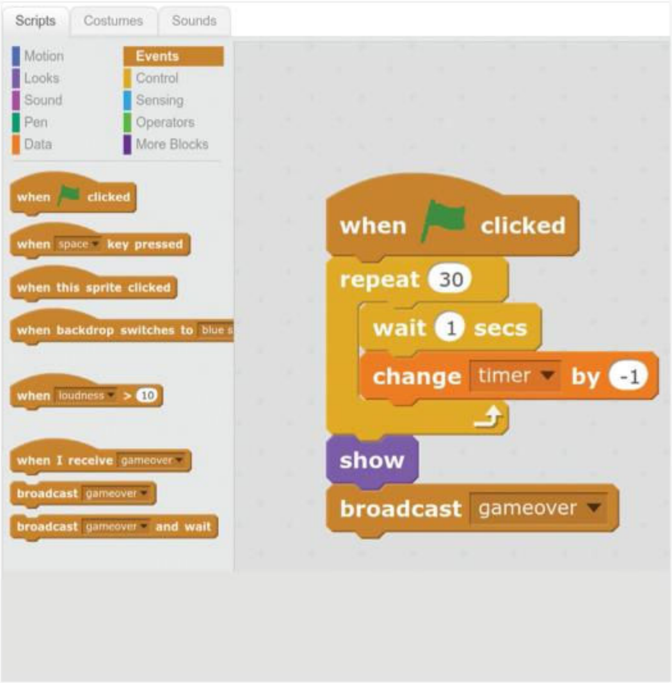
Variables store data, and we'll create two to store our score and timer data. In the Data panel (the new name for "Variables") click on "Make a Variable", name it "score" and make it available for all sprites. Now repeat this step to create another variable called "timer". Once done, next create the blocks shown, which will reset the score and timer when the game starts.



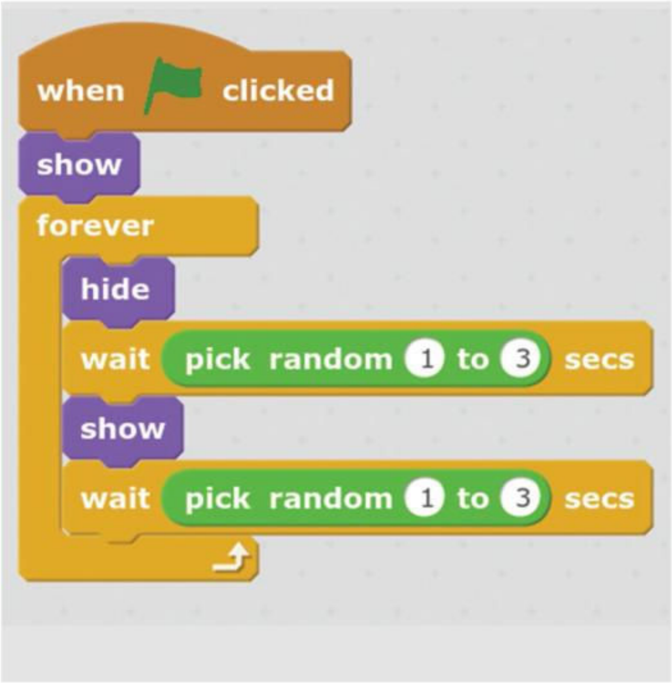
```
when [green flag] clicked
repeat 30
    wait 1 secs
    change timer by -1
show
```

6 Create a timer

Our timer uses a "repeat" loop from the Control section. We'll need to edit this loop in order to repeat our code 30 times, thus giving us a 30-second timer. Inside the loop, therefore, we wait for one second, then deduct 1 from the "timer" variable, which effectively creates a countdown. Outside of the loop we use "show" from Looks to ensure that our sprite is visible once the timer ends.


Scripts
Costumes
Sounds
Motion
Looks
Sound
Pen
Data
Events
Control
Sensing
Operators
More Blocks
when ⚑ clicked
when space key pressed
when this sprite clicked
when backdrop switches to blue s
when loudness > 10
when I receive gameover
broadcast gameover
broadcast gameover and wait

```
when ⚑ clicked
repeat 30
    wait 1 secs
    change timer by -1
show
broadcast gameover
```



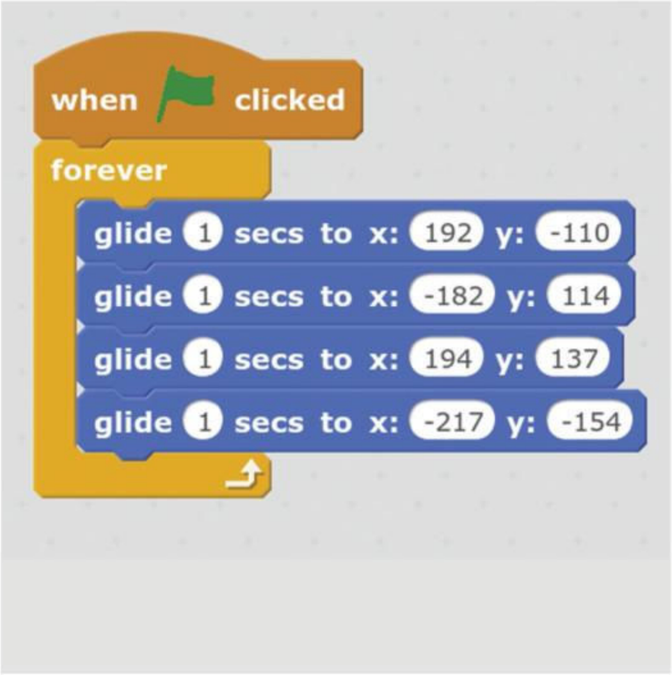
```
when ⚑ clicked
show
forever
    hide
    wait pick random 1 to 3 secs
    show
    wait pick random 1 to 3 secs
```

7 Broadcast yourself

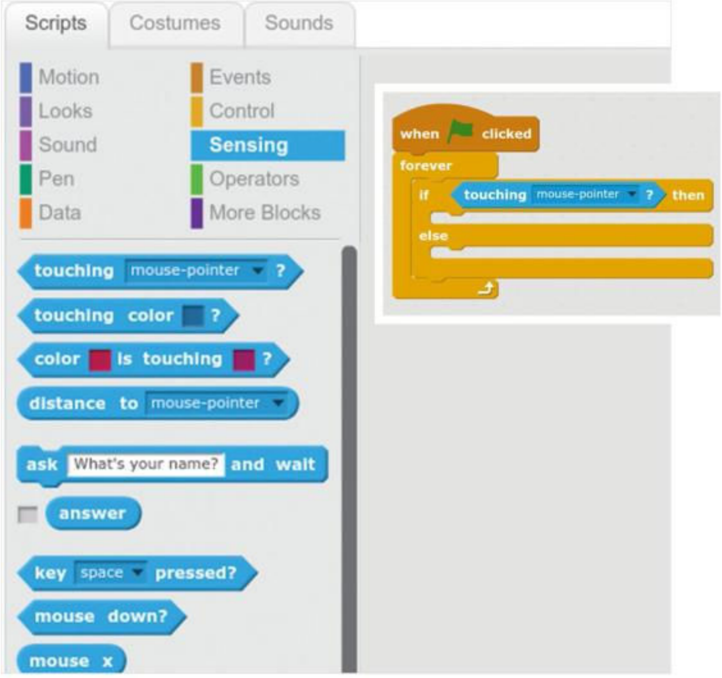
Underneath the code for our timer, but still attached, we create a new block, a broadcast. As we mentioned last time, broadcasts enable sprites to communicate with each other. Broadcasts are found in Events. Click on the drop-down for "broadcast" and create a new message called "gameover", then connect it to our timer code just underneath the show block.

8 Hiding our duck

Our duck can disappear from view thanks to this sequence of code. We first ensure that the duck is visible, using "show". Next we use a "forever" loop from Control and inside we add a hide and show block. But in between these blocks we place a "wait 1 secs" block from Control and then place "pick random" from Operators over the "1" and alter accordingly.



```
when ⚑ clicked
forever
    glide 1 secs to x: 192 y: -110
    glide 1 secs to x: -182 y: 114
    glide 1 secs to x: 194 y: 137
    glide 1 secs to x: -217 y: -154
```


Scripts
Costumes
Sounds
Motion
Looks
Sound
Pen
Data
Events
Control
Sensing
Operators
More Blocks
touching mouse-pointer ?
touching color ?
color is touching ?
distance to mouse-pointer
ask What's your name? and wait
answer
key space pressed?
mouse down?
mouse x

```
when ⚑ clicked
forever
    if touching mouse-pointer ? then
    else
```

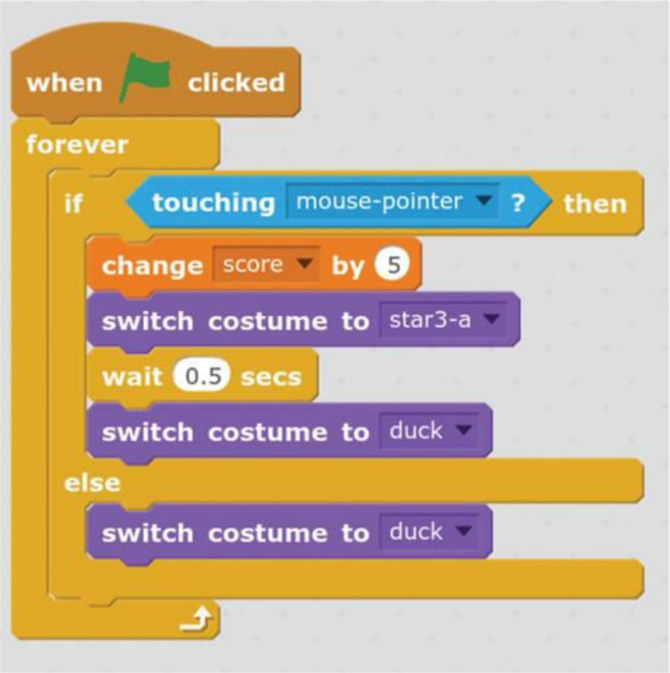
9 A moving target

A sitting duck is no fun, so we'll give it a bit of life. Using another "forever" loop we place four "glide" blocks from the Motion category, which will deal with the motion of the sprite. The x and y coordinates are generated by moving the duck around to places where you want it to go. This updates the coordinates in the block, which you then drag into the loop, one by one.

10 Conditions are True

Using a forever loop and an If condition, we can now have our code check to see if our duck sprite is touching the mouse pointer – you can find this block in the Sensing category. Note that the shape of the block is a hexagon, and so is the condition to test in our conditional loop. Drag the block into the hexagon space and ensure that it says "mouse-pointer".

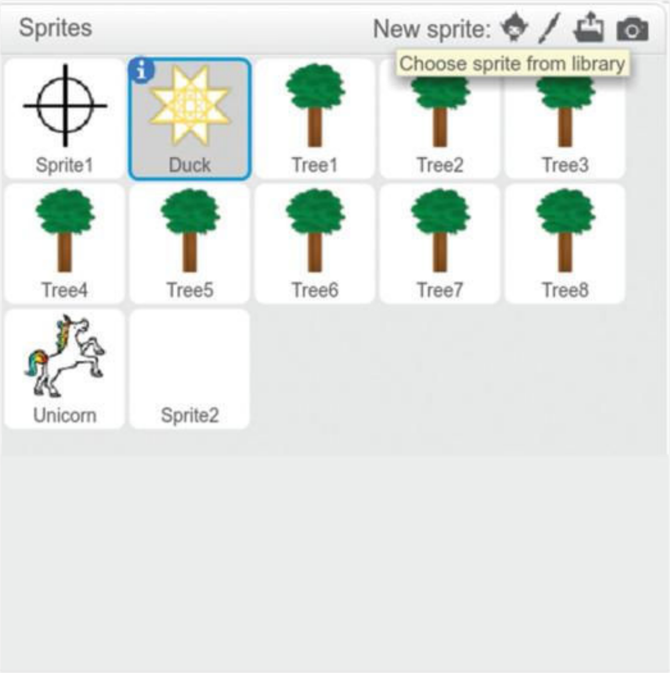
»



```
when [green flag] clicked
forever
    if touching mouse-pointer ? then
        change score by 5
        switch costume to star3-a
        wait 0.5 secs
        switch costume to duck
    else
        switch costume to duck
```

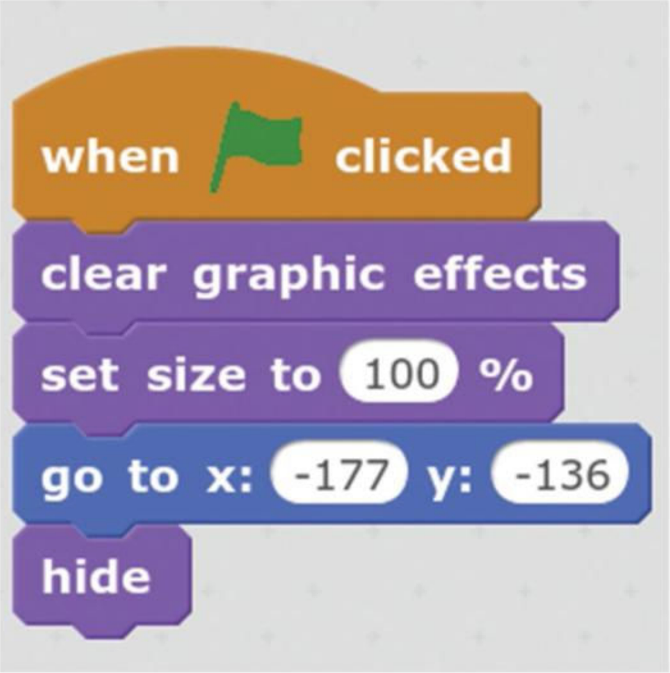
11 Costume change

Inside of our "If" loop we now use a "change" block from the Data category to alter our score by five points every time we target the duck. We then change our costume using the "switch costume" block from Looks. We briefly change to a star pattern explosion, then change back. Inside the else section we ensure that our duck remains in its normal costume.



12 Adding a new sprite

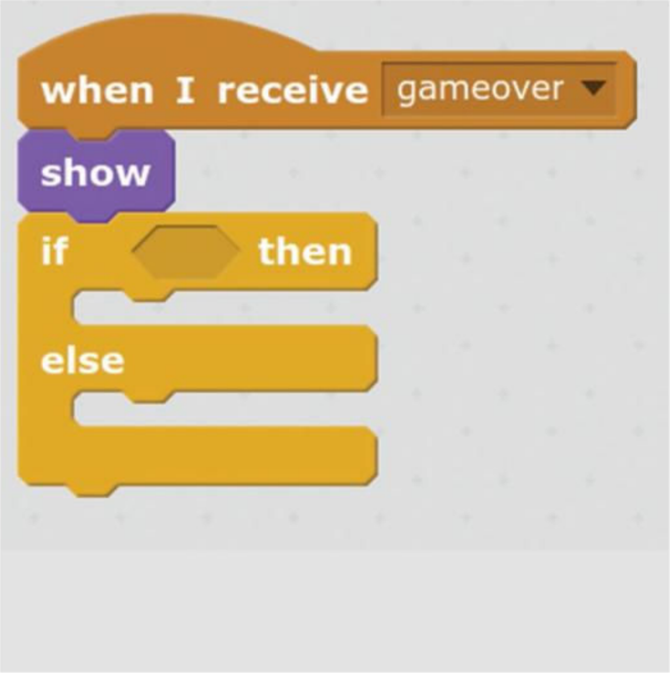
Now we create a new sprite. Look for the New Sprite text under the Stage and click on the New Sprite icon to load the library. We chose a Unicorn as our sprite to announce the score at the end of the game. Feel free to choose any sprite that you wish.



```
when [green flag] clicked
clear graphic effects
set size to 100 %
go to x: -177 y: -136
hide
```

13 Unicorn powers ACTIVATE!

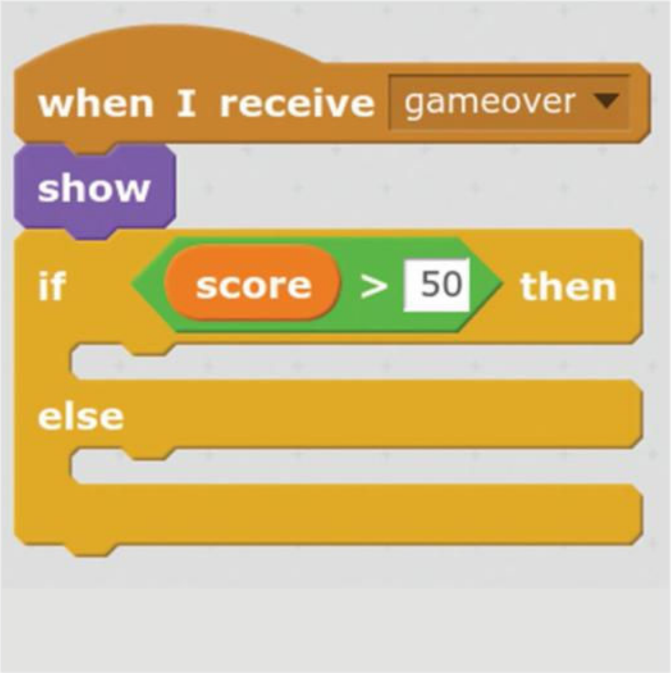
Our Unicorn must disappear when the game starts – when the Green Flag is clicked – so we use blocks from Looks to clear any special effects that we will later use on the sprite, then set its size to normal. Then we ensure that the Unicorn is in the bottom-left of the screen using the "go to" block from Motion. Finally we hide the Unicorn.



```
when I receive gameover
show
if < > then
else
```

14 Broadcast received...

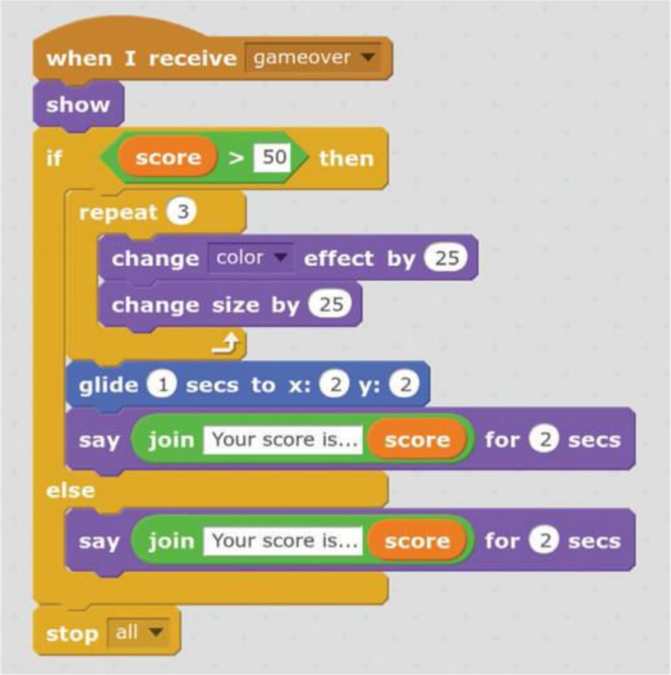
Remember the "gameover" broadcast that we created earlier? Well, now we use it. In Events look for "when I receive.." and drag it into the coding area. Then drag another "show" block from Looks and connect it under the previous block. We then need to drag an "If.. Else" block from Control and place it under our previous blocks.



```
when I receive gameover
show
if score > 50 then

else

```

15 Conditional Rewards

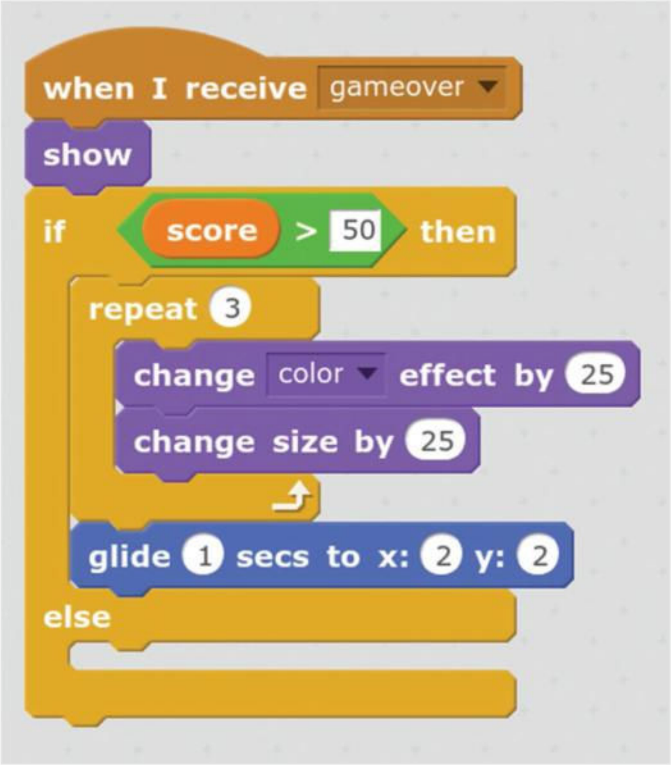
Using a comparison block from Operators, we check that the value stored in our "score" variable is greater than 50. (We chose this value to give the player a chance to get a rewarding end result, but you can change this as you see fit...) If the player's score is greater than 50 then we go into the "If" condition code.



```
when I receive gameover
show
if score > 50 then
    repeat 3
        change color effect by 25
        change size by 25
    glide 1 secs to x: 2 y: 2
    say join Your score is... score for 2 secs
else
    say join Your score is... score for 2 secs
stop all
```

17 Announcing our score

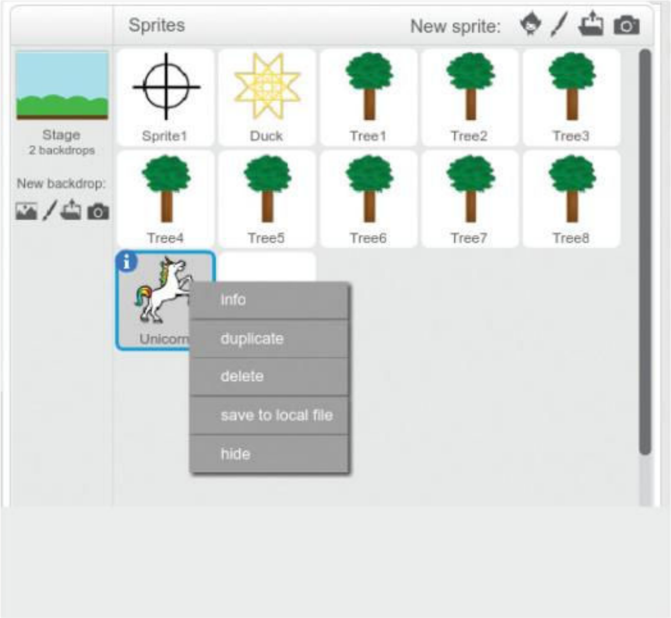
No matter what the player's score is, the Unicorn will announce it by using a "say" block from Looks. But we replace the text with "join" from Operators and change the first block of text. For the second block of text we drop our "score" variable, from Data, inside the space. Lastly we use "stop all" from Control to stop the game.



```
when I receive gameover
show
if score > 50 then
    repeat 3
        change color effect by 25
        change size by 25
    glide 1 secs to x: 2 y: 2
else

```

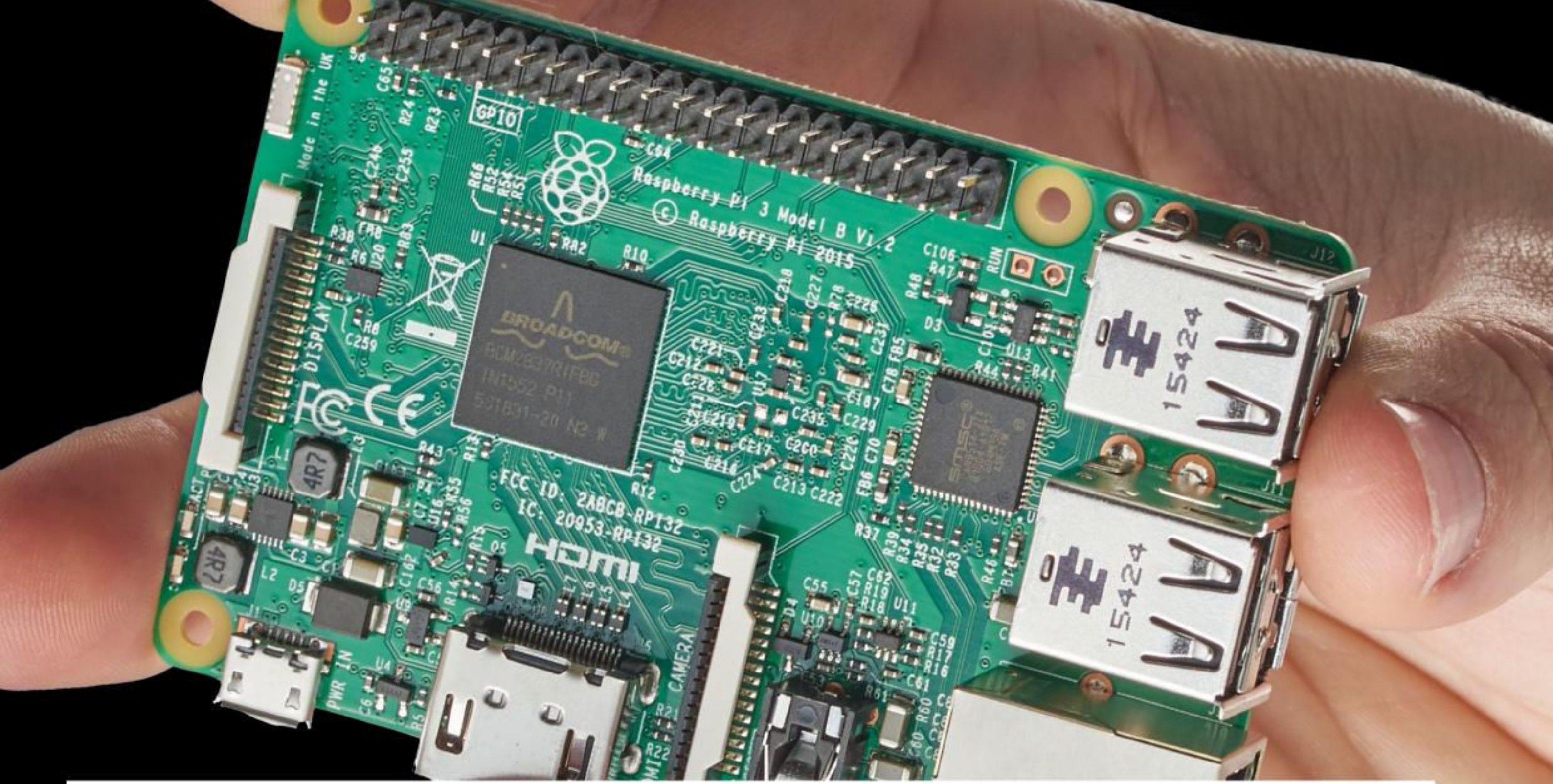
16 Colour changing giant Unicorn

Unicorns are famous for looking fabulous, so using a repeat loop from Control and two blocks from Looks, we change the colour effect, cycle through colours, and change the size of the sprite by 25 percent. This happens three times before the loop ends and the Unicorn gracefully glides to the centre of the screen.



18 Adding a forest

We added another sprite in the shape of a tree, and then used it to provide cover for our duck. To duplicate a sprite, right-click on the sprite and select Duplicate. We also clicked on the stage icon, to the left of the panel, and changed the backdrop in the same way that we changed costumes for our duck sprite. ■



First steps in Python: Control structures

Mayank Sharma takes your fledgling Python skills further by introducing new concepts and helping you write your first Python script

We introduced two types of variables in our introduction to Python in the previous issue: integers and strings. That's actually a little misleading since unlike in other programming languages Python variables do not need explicit declaration. Instead this happens automatically when you assign a value to a variable. Besides whole numbers and a collection of characters, you can also store decimal numbers and boolean values inside variables.

Every piece of data has to have a type associated with it so Python knows what it's dealing with. You can find out what type Python has associated with a bit of data by using the **type()** function. In a Python interpreter, enter:

```
>>> type (3.5)
<type 'float'>

>>> type ("Quotes make it a string")
<type 'str'>

>>> type (1>3)
<type 'bool'>
```

Boolean or bool is the simplest type of variable because there are just two values it can take: True or False. Unlike other variable types, however, a bool value isn't usually stored explicitly in variables but is rather used in conditions for loops and if statements, which we'll cover in more detail a little later.

As well as these simple data types, Python allows us to bring data together in various ways to create structures. The simplest structures are called sequences, which store information one after the next. There are two sorts of sequences, called lists and tuples. In many ways, they're very similar. A list contains a collection of data in a specific order while a tuple contains a collection of immutable data in a specific order. This means that while you can update an item in a list you cannot do so in a tuple.

You create a list by enclosing it in square brackets, while tuples are surrounded by round ones (parentheses). You can retrieve an individual item from either by putting square brackets after it along with an index number. Note that indexes start at 0, so `list_1[0]` and `tuple_1[0` will retrieve the first item in either sequence.

Entrance fee

Now let's work on a program to demonstrate how to use variables and different data types and also understand the important concepts of conditional statements and loops.

A particular zoo in London determines the price of admission based on the age of the visitor. Visitors 3 years of age and under are charged a

Pi bites

The Pi uses only around 5-7 watts of power. The consumption varies with how you use it – 1.2A will be enough for most tasks, though it needs 2.5A to use all four USB ports.

nominal £0.99 fee. Children between 3 and 15 years of age are charged £17.25. Seniors aged 60 years and over get a concession and are charged £19.35. All other visitors are charged as adults at £21.50.

We'll create a program that'll read the ages of all the visitors in a group and add up their admission fees based on the above given criteria. When the user enter **0** to indicate that there are no more visitors in the group, the program will then display the admission cost for the group.

First up we'll store the different admission prices and age limits in variables, such as:

```
# The various admission prices
baby_price = 0.99
child_price = 17.25
adult_price = 21.50
senior_price = 19.35

# The applicable age limits
baby_age_limit = 3
child_age_limit = 15
adult_age_limit = 60

# Variable to hold the total admission price
total = 0
```

In this block of code, we've first stored the applicable admission prices as floating point numbers and the age limits as integers. We've also created a variable that'll hold the final computed admission price. The line beginning with **#** is a comment that explains the code. Any characters on the same line that follow the # symbol will be ignored by Python.

Next up we'll ask the user to enter the age of a person in the group:

```
visitor = input ("Enter the age of the visitor (0 to
exit): ")
```

Python provides the **input()** function, which has an optional prompt string parameter. When the input function is called, the program flow will be stopped until the user has given an input and has ended the input with the Return key. First it prints the string you give as a parameter, and then it waits for a line to be typed in. In the line above, the entered value is assigned to the variable named **visitor**, which we'll use later.

Note that while the parameter inside the parentheses after input is optional, it is very important since it prompts you that the program is waiting for you to enter something from the keyboard. Without the prompt, Python would just sit there waiting and the user would not know what was happening.

Control structures

A program is usually executed linearly from top to bottom, one line at a time. However, certain statements, referred to as control-flow statements, have the power to change the path of the code execution.

"While" loops are one of the most common control-flow statements. The "while" loop has a condition that can be anything with a bool data type, and the code block will continue looping until that condition is False. The condition can be as complicated as you like, as long as it comes back to either True or False. However, in order for the loop to eventually terminate, the code block must include one or more variables that can change inside the loop.

Secondly, many programs require you to go through more than one thing at a time, such as in our case where we have to calculate totals based on multiple ages. One of the simplest ways to control the flow of a Python program is to branch off and execute different pieces of code depending on a condition. This is best done with the "if" loop. Like "while" loops, "if" statements take a condition that has to have the bool data type. Also, "if" statements only execute one block of code. Once Python finds a condition that is True, it'll execute that block and then exit the "if" statement.

That said, "if" statements can also have an "else" statement and additional "elif" statements. Of these, "elif" is used to create an extra set of conditions in the if-else statement. The "elif" clause is a combination of the "else" clause and a separate "if" statement. When an "if" condition is False, Python skips the indented block and looks for an "elif", which it then evaluates. Once again, if this is determined to be True, Python then executes the following block. When the "elif" condition turns False, Python then moves forward and looks for another "elif" or an "else". The "else" block is executed if none of the conditions are True. Note that "if" blocks don't have to have an "elif" or "else" statements. If there's no "else" statement, and none of the conditions are True, then Python simply exits the "if" block without executing anything.

»

Pi bites

100,000 Raspberry Pis were sold on the first day of launch in February 2012, and more than 10 million units have been sold worldwide as of September 2016.

Programming languages vs. scripting languages

While people might squabble over it, Python is strictly speaking a scripting language. A scripting language differs from a true programming language in a couple of ways. The foremost difference between the two is that programming languages are compiled, unlike scripting languages. Languages like C, C++, and Java must be compiled by a compiler, which then creates a file of machine code that is unreadable by humans. For example, when you write a program in C and compile it, the resulting .o file is what is read by the computer. Scripting languages, on the other hand, are read, interpreted and acted upon every time you run them.

There are other differences as well. Programming languages most often run directly on top of the hardware on which they are written, though Java is an exception since it runs inside the Java Virtual Machine, which is what makes it so portable. Also, programming languages tend to be more complex and difficult to learn. This is in fact one of the reasons for the popularity of programming in Python on the Raspberry Pi. Together they lower the entry threshold and make programming accessible to people who would never have dreamt of programming a computer.

Naming variables

Python gives you a wide latitude to assign a name to your variables as long as you follow certain rules. The variable name can include numbers, letters and special characters, as long as the first character isn't a number or the underscore character. Also out of bounds are words that Python uses itself, such as if, for, while. Your variable names also must be single words and can't contain spaces. Some people prefer to create verbose variable names such as **vendor_number** while some prefer abbreviated versions like **vnum**. Similarly, many people use the underscore character to separate the words in their variable names while others use camelCase, where each word begins with a capital letter. You can use any of these practices as long as the names are readable. Also, irrespective of the system you use, always make sure you are consistent.

All this might sound a bit overwhelming at first but it really isn't once you get used to it. Let's put this knowledge to work to compute the entrance fee for a group of people. We'll start off with a "while" loop:

```
while visitor != 0:
    age = int(visitor)
```

The visitor variable holds the age of a visitor as entered by the user. Python will stay within this loop until the user enters 0 as the age of the visitor (or rather, enters this to indicate that there are no more visitors to consider), at which point the condition for the "while" loop turns False and Python skips the indented "while" block. Once inside the "while" loop, we call upon Python's **int()** function to convert the entered age into a plain integer. The converted age is then stored inside another variable called **age**.

Now comes the crucial bit:

```
if age <= baby_age_limit:
  total = total + baby_price
elif age <= child_age_limit:
  total = total + child_price
elif age <= adult_age_limit:
  total = total + adult_price
else:
  total = total + senior_price
```

These few lines are the meat of the program that do all the heavy lifting. The "if" and "elif" lines check whether the age falls within one of the predefined age limits. If the age is less than or equal to the age defined in the **baby_age_limit** variable, then the price of their admission is added to the current value of the total variable. If this condition is false, Python looks for an "elif" condition and then evaluates it. Once it has evaluated all the conditions and none of them hold true, it then enters the "else" block.

The "else" block is triggered only when the entered age doesn't match any of the defined age limits. This will happen only when the age is over 60. To make it simpler, we could have replaced the "else" statement with an "elif" to check if the age meets the criteria for the senior price, like this:

```
elif age > adult_age_limit:
  total = total + senior_price
```

Pi facts

The credit-card-sized Pi is so compact and well designed that it has no moving parts and hence doesn't produce any noise or vibrations.

Python uses a colon to indicate the start of a block of code and then continues until the indentation changes. In practice what this mean is that you add a colon and then hit the Enter key to start a new line. You then hit Tab once and then type your first line. Every line that's also tabbed will then be executed one after the other.

Variable formatting

To keep things moving forward, we'll ask the user to enter the age for the next visitor in the group. We'll copy the line that we used earlier to enter the "while" loop. This one refreshes the value in the age variable to evaluate whether the "while" loop still holds true:

```
visitor = input ("Enter the age of the visitor (0 to exit): ")
```

But what happens when the user enters 0 and the condition for the "while" loop becomes False? In that case, we'll exit the loop and display the value of the total variable, which now holds the admission price for the entire group.

```
print "The total entrance fee for the group is %.2f" %total
```

Displaying values from variables in Python seems a little bit odd at first. The **f** following the first **%** is short for float. The **.2** tells Python to display the value in the floating point variable with two decimal places. The left **%** tells Python where you want to put the formatted string. The value following the right **%** is the variable that we want to display. Python takes the value following the second **%** and formats it according to the first **%**. Then it places the second value in place of the first **%**. The above line will produce a string that reads something like this:

```
The total entrance fee for the group is 56.94.
```

We'll look at Python formatting in more detail in our next tutorial in the next issue.

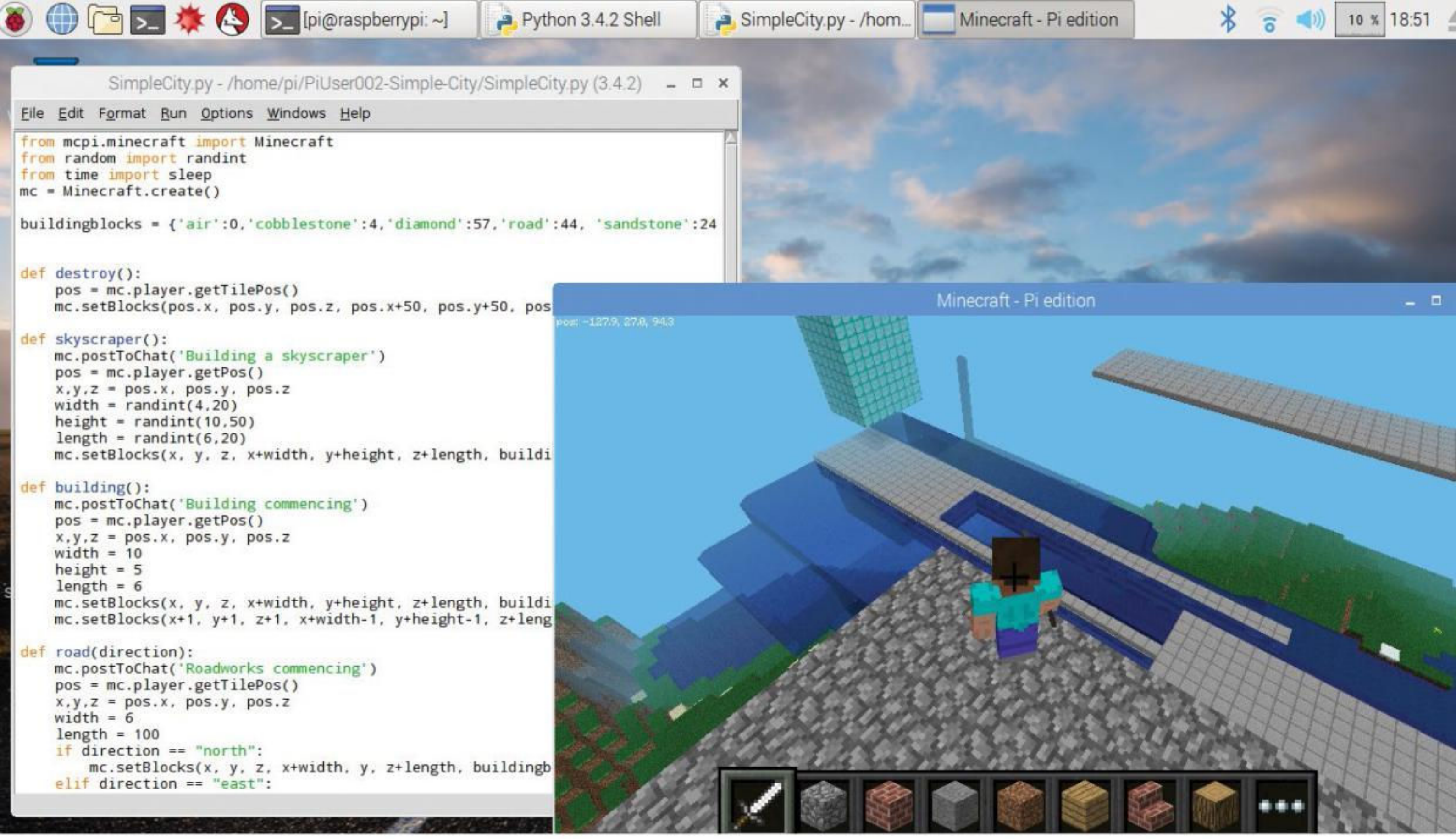
Head to **https://github.com/geekybodhi/piuser/blob/master/entrance-fee.py** to grab the complete code for the admission example used in this tutorial. The code includes hints to also calculate the number of individuals in a group. As an exercise, extend this example to record and then list the number of visitors in the group based on their ages. ■

Jargon buster

Variable: A named storage location that is used to hold information which will be referenced and in all likelihood be manipulated in the program.

Function: A block of reusable code that is called upon to perform a specific task. Python comes with a predefined set of functions, which are stored in a library.

Parameter: Also known as an argument, this is a value that is passed to a function. A function can have multiple parameters, and they can be either required or optional.



```
from mcpi.minecraft import Minecraft
from random import randint
from time import sleep
mc = Minecraft.create()

buildingblocks = {'air':0,'cobblestone':4,'diamond':57,'road':44, 'sandstone':24


def destroy():
    pos = mc.player.getTilePos()
    mc.setBlocks(pos.x, pos.y, pos.z, pos.x+50, pos.y+50, pos

def skyscraper():
    mc.postToChat('Building a skyscraper')
    pos = mc.player.getPos()
    x,y,z = pos.x, pos.y, pos.z
    width = randint(4,20)
    height = randint(10,50)
    length = randint(6,20)
    mc.setBlocks(x, y, z, x+width, y+height, z+length, buildi

def building():
    mc.postToChat('Building commencing')
    pos = mc.player.getPos()
    x,y,z = pos.x, pos.y, pos.z
    width = 10
    height = 5
    length = 6
    mc.setBlocks(x, y, z, x+width, y+height, z+length, buildi
    mc.setBlocks(x+1, y+1, z+1, x+width-1, y+height-1, z+leng

def road(direction):
    mc.postToChat('Roadworks commencing')
    pos = mc.player.getTilePos()
    x,y,z = pos.x, pos.y, pos.z
    width = 6
    length = 100
    if direction == "north":
        mc.setBlocks(x, y, z, x+width, y, z+length, buildingb
    elif direction == "east":
```

Building your own cities in Minecraft

Les Pounder shows you how some simple Python code can enable anyone to create – and destroy – their own city in Minecraft

Back in the 1980s an ambitious game saw players building and running their own city: SimCity. It has been ported to many different machines, from ZX Spectrum to PC and everything in between, but the basic game element is to create your own sprawling city out of nothing. You have godlike powers to do and build whatever you wish, but will you use them for good or for evil?

Well, we'll use some Python code and Minecraft to create our own SimpleCity, which will shape the Minecraft world into our vision of the future. All of the code for this tutorial can be downloaded from **https://github.com/lesp/PiUser002-Simple-City/archive/master.zip**

Coding the project

The goal of our project is to create an easy-to-use menu that will enable the player to quickly build their own city in the Minecraft world. To start, open the Python 3 editor from the main Programming menu. Once it's open you will see the Python shell. Click File > New to create a new file. You now see a blank document. Click File > Save and save the project as "SimpleCity.py". Subsequent saves will now take less time. Remember to save often!

With our blank document open we start writing the Python code. We first import a series of modules – code libraries that enable Python to do more. First is the Minecraft library, which enables our code to talk to Minecraft. Next import one class from the random library: randint, short for random integer. Finally import the sleep class from the time library, used to add delays to our code.

```
from mcpi.minecraft import Minecraft
from random import randint
from time import sleep
```

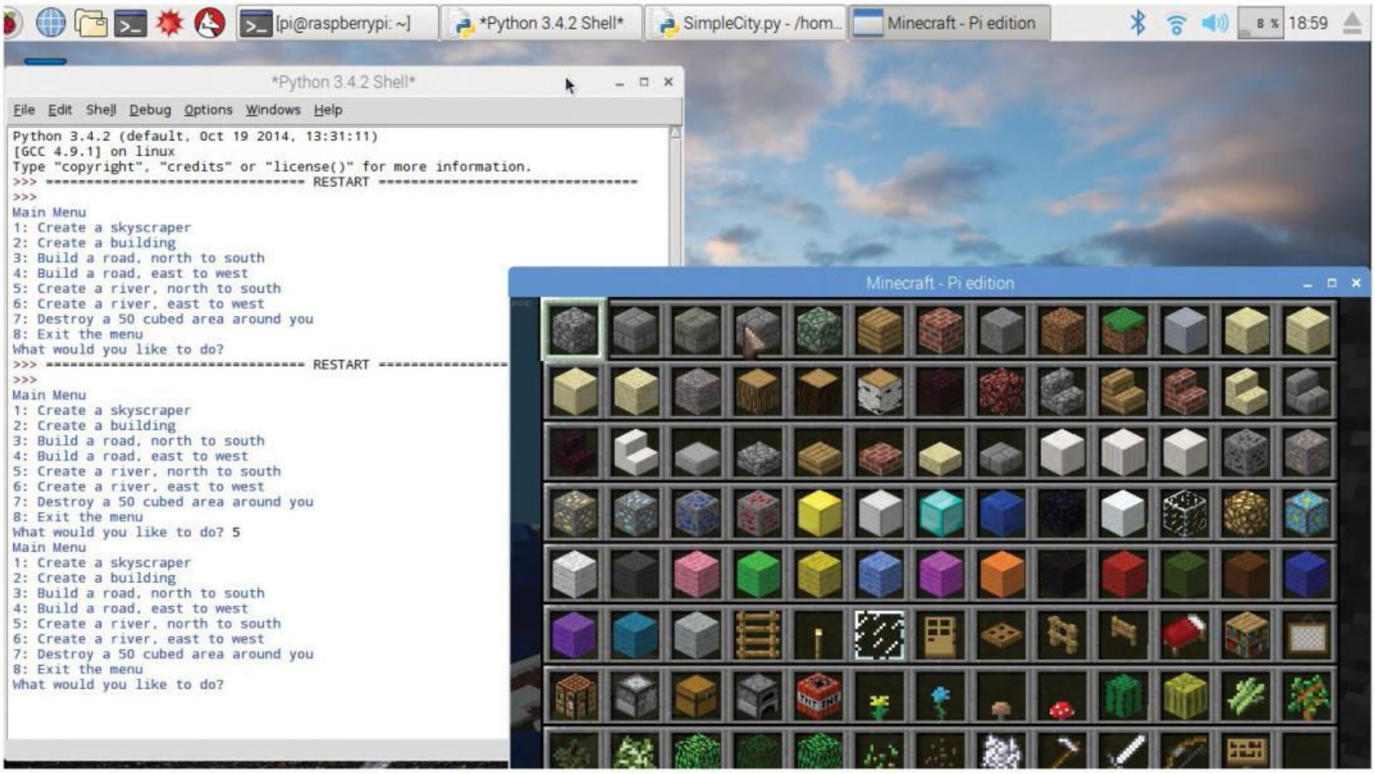
Next we create an object that will create a connection from our code to Minecraft:

```
mc = Minecraft.create()
```

Our next line of code is a dictionary. A dictionary can be used to store multiple items, which can then be referenced and used by calling their name, otherwise known as a "key". For example if we use the key **air** then we return the value **0**. Here we use a dictionary to store the block IDs for a selection of blocks that we use in the project.

```
buildingblocks = {'air':0, 'cobblestone':4,
'diamond':57, 'road':44, 'sandstone':24, 'water':8}
```

We now start a section where we create a series of functions, which is a block of reusable code that we use to handle building or destroying sections of »



```
Python 3.4.2 (default, Oct 19 2014, 13:31:11)
[GCC 4.9.1] on linux
Type "copyright", "credits" or "license()" for more information.
>>> ================================ RESTART ================================
>>>
Main Menu
1: Create a skyscraper
2: Create a building
3: Build a road, north to south
4: Build a road, east to west
5: Create a river, north to south
6: Create a river, east to west
7: Destroy a 50 cubed area around you
8: Exit the menu
What would you like to do?
>>> ================================ RESTART ================
>>>
Main Menu
1: Create a skyscraper
2: Create a building
3: Build a road, north to south
4: Build a road, east to west
5: Create a river, north to south
6: Create a river, east to west
7: Destroy a 50 cubed area around you
8: Exit the menu
What would you like to do? 5
Main Menu
1: Create a skyscraper
2: Create a building
3: Build a road, north to south
4: Build a road, east to west
5: Create a river, north to south
6: Create a river, east to west
7: Destroy a 50 cubed area around you
8: Exit the menu
What would you like to do?
```

The inventory contains lots of extra blocks and items that you can use to decorate your home and the city around it.

the Minecraft world. We start our functions by creating a block of code that will destroy a 50-cubed area of the world. First we give the function a name: "destroy". Notice the colon at the end of the line – this tells Python that the next line should be indented. Our editor will automatically do this for us. The first line of indented code creates a variable called **pos** that will store the position of the player in the world using the **getTilePos** command, which returns a coarse but adequate position for placing blocks. Once we know the player's position, we next set the blocks, using **setBlocks**, from the player's position (x,y,z) to 50 blocks away for each axis. This creates a 50 cubed box containing air, effectively destroying everything in that cube.

```
def destroy():
    pos = mc.player.getTilePos()
    mc.setBlocks(pos.x, pos.y, pos.z, pos.x+50, pos.y+50, pos.z+50, buildingblocks['air'])
```

Our next function creates rather than destroys. It creates a Skyscraper at the player's position. First it will post to the Minecraft chat screen that building has started. We then repeat the process to locate the player in the world, but then we split the data stored in "pos" into separate x,y,z variables. The next three lines create variables **width**, **height** and **length**, each with a randomly generated number as its value. We use **randint** and specify a range in which the number can be selected. In the last line we set the blocks for an area starting from the player's position and ending at the randomly chosen coordinates. We used a diamond block to give our skyscraper a shimmering tower of glass.

```
def skyscraper():
    mc.postToChat('Building a skyscraper')
    pos = mc.player.getPos()
    x,y,z = pos.x, pos.y, pos.z
    width = randint(4,20)
    height = randint(10,50)
    length = randint(6,20)
    mc.setBlocks(x, y, z, x+width, y+height, z+length, buildingblocks['diamond'])
```

As well as skyscrapers we need a simple home for the people who live in the city, and our next function, **building**, will build a modest sized home, ready for you to decorate. Building works in the same way as our skyscraper function, but it builds only homes of a certain size, and out of cobblestone. The key difference with this function is that after we create the cobblestone that makes the building, we create another smaller space within by creating a building one cube smaller in every axis, out of air, essentially hollowing out the home. We will leave doors and windows to you – just remember to press "e" on the keyboard to access the inventory for your belongings.

```
def building():
    mc.postToChat('Building commencing')
    pos = mc.player.getPos()
    x,y,z = pos.x, pos.y, pos.z
```

SNAKE CHARMER

Python is a wonderful language. In this project we have used it to conjure up new worlds and give us godlike powers. But the true power of Python on the Raspberry Pi is that we can use any Python library in our projects. This includes the GPIO Zero library, which enables easy use of the Pi's 40 GPIO (General Purpose Input/Output) pins. So we can now have Minecraft react to the physical world. For example in our project, the functions that create roads, rivers and skyscrapers are triggered using a menu, but we can also use buttons, sensors and triggers in the real world to invoke these functions. Think of destroying a 50-cubed area of the world just by waving your hand in front of a sensor – true godlike powers can be yours with Python! You can find out more about GPIO Zero by reading the official documentation at **http://bit.ly/gpio-zero**

```
    width = 10
    height = 5
    length = 6
    mc.setBlocks(x, y, z, x+width, y+height,
z+length, buildingblocks['cobblestone'])
    mc.setBlocks(x+1, y+1, z+1, x+width-1,
y+height-1, z+length-1, buildingblocks['air'])
```

We have our buildings, now we need roads. This function is a little different, in that it takes an "argument" – in other words, we give the function an extra piece of information. In this case we tell the function which direction to build the road in. The function replicates the chat message to the user, gets the position of the user, and sets the width of the road to 6 blocks and the length to 100 (these can be changed to suit your needs). We then enter into a conditional test that will check the direction given by the player. This uses two conditions, the first being **if** and the second **elif**, short for "else if". If the direction is "north" then the road is built ahead of the player. Else if the player types "east" then the road is built across their path.

```
def road(direction):
    mc.postToChat('Roadworks commencing')
    pos = mc.player.getTilePos()
    x,y,z = pos.x, pos.y, pos.z
    width = 6
    length = 100
    if direction == "north":
        mc.setBlocks(x, y, z, x+width, y, z+length,
buildingblocks['road'])
    elif direction == "east":
        mc.setBlocks(x-1, y-1, z-1, x+length,
y,z+width, buildingblocks['road'])
```

Next we create the function that will build rivers. This is essentially the same as our road function, including the ability to pass an argument, direction, so that we can build roads in front of our path or across our path. But the river code differs in that we can also set the depth of the water. We chose 6 blocks depth as it is three times the height of the player, just enough for the player to sink but not too deep that we get lost. Our player cannot drown, so you are free to explore the underwater world. When creating a river it is wise to remember that water flows, and that if the river is created somewhere high, say a hill, then water will cascade and flood the ground – but this can be used to your advantage if you'd like a skyscraper waterfall!

```
def river(direction):
    mc.postToChat('Can I swim?')
    pos = mc.player.getTilePos()
    x,y,z = pos.x, pos.y, pos.z
    width = randint(1,5)
    depth = 6
    length = randint(1,100)
    if direction == "north":
            mc.setBlocks(x, y, z, x+width, y-depth,
z+length, buildingblocks['water'])
    elif direction == "east":
            mc.setBlocks(x, y, z, x+length, y-depth,
z+width, buildingblocks['water'])
```

Our functions are now complete and we move on to the next section of code. Here we create two lines that use the chat window to welcome the player and advise them on what to do in the game.

```
mc.postToChat('Welcome to our Simple City.')
mc.postToChat('You are free to destroy and create
using the menu')
```

Next we create a variable called "answer" and into it we store the integer 0:

```
answer = 0
```

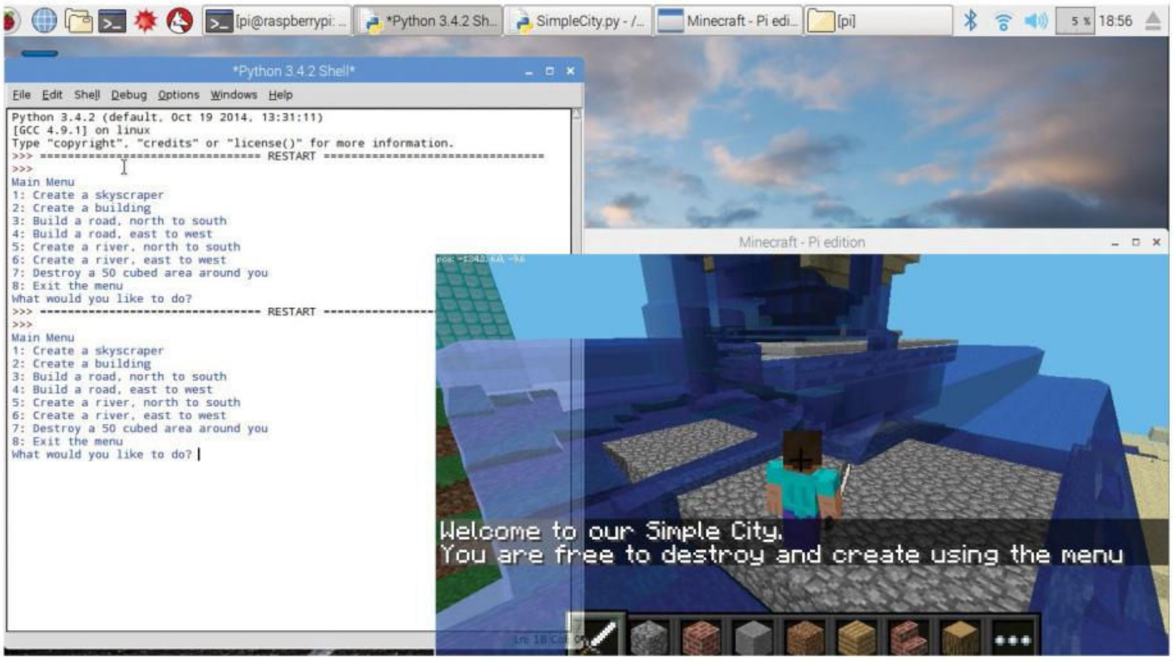
Why do we do this? Well, in order for us to use a "while answer" loop, we first need to create the "answer" variable, otherwise we cannot enter the loop. This while loop will compare the value stored in "answer" and if it does not equal the value "8" then the code in the loop will run.

But where does this value come from? Well, we create a menu, which has a number attached to a function. This menu is a multi-line print function, which is neater than creating a print function for each line. You will notice that at the end of each

»

Pi bites

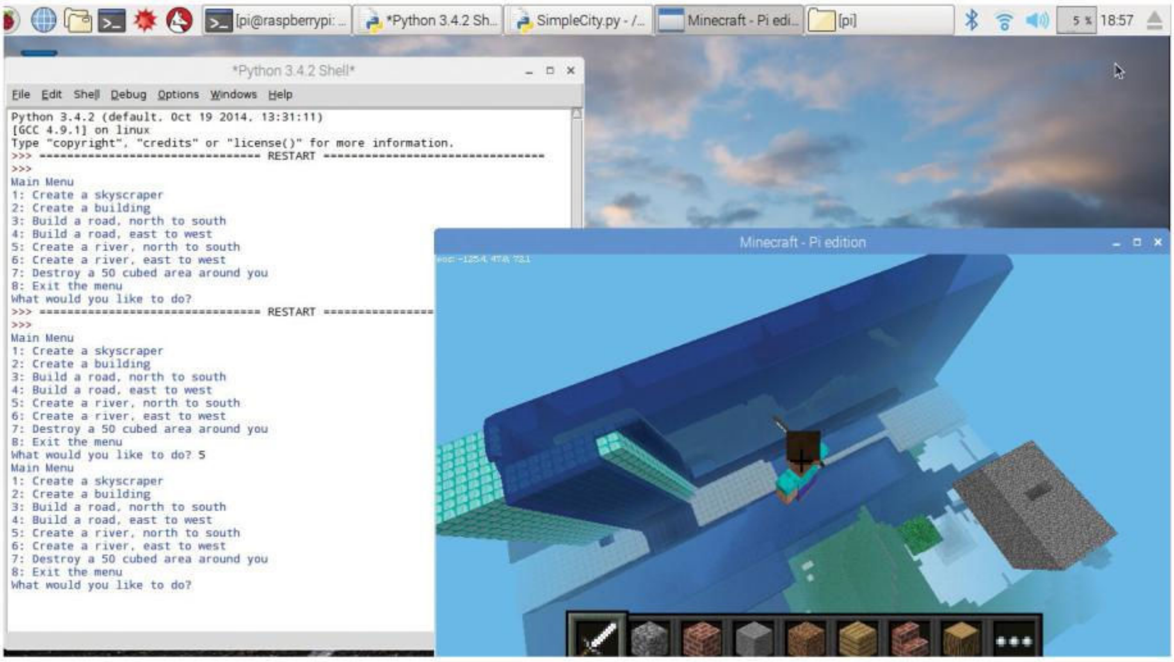
In Minecraft Pi, if the player tries to play full screen, they cannot access all items in the inventory. So always play Minecraft in a window.



```
Python 3.4.2 (default, Oct 19 2014, 13:31:11)
[GCC 4.9.1] on linux
Type "copyright", "credits" or "license()" for more information.
>>> ================================ RESTART ================================
>>>
Main Menu
1: Create a skyscraper
2: Create a building
3: Build a road, north to south
4: Build a road, east to west
5: Create a river, north to south
6: Create a river, east to west
7: Destroy a 50 cubed area around you
8: Exit the menu
What would you like to do?
>>> -------------------------------- RESTART ---------------
>>>
Main Menu
1: Create a skyscraper
2: Create a building
3: Build a road, north to south
4: Build a road, east to west
5: Create a river, north to south
6: Create a river, east to west
7: Destroy a 50 cubed area around you
8: Exit the menu
What would you like to do? |
```

```
Welcome to our Simple City.
You are free to destroy and create using the menu
```

« **Starting our project launches a simple menu in the Python shell and a quick message pops up in Minecraft.**



```
Python 3.4.2 (default, Oct 19 2014, 13:31:11)
[GCC 4.9.1] on linux
Type "copyright", "credits" or "license()" for more information.
>>> ================================ RESTART ================================
>>>
Main Menu
1: Create a skyscraper
2: Create a building
3: Build a road, north to south
4: Build a road, east to west
5: Create a river, north to south
6: Create a river, east to west
7: Destroy a 50 cubed area around you
8: Exit the menu
What would you like to do?
>>> ================================ RESTART ===============
>>>
Main Menu
1: Create a skyscraper
2: Create a building
3: Build a road, north to south
4: Build a road, east to west
5: Create a river, north to south
6: Create a river, east to west
7: Destroy a 50 cubed area around you
8: Exit the menu
What would you like to do? 5
Main Menu
1: Create a skyscraper
2: Create a building
3: Build a road, north to south
4: Build a road, east to west
5: Create a river, north to south
6: Create a river, east to west
7: Destroy a 50 cubed area around you
8: Exit the menu
What would you like to do?
```

❯ **Our city has a massive waterfall that cuts into a skyscraper. Water is extremely damaging in Minecraft, so use with caution.**

line is **\n** – this is an escape character and tells Python to create a new line. After the print function we update the "answer" variable with input from the user. Here the user is asked to choose a number, which is then stored in the variable. For example if we wish to build a skyscraper we press **1** and then Enter. This is captured and stored in "answer" ready for the next section of code.

```
while answer != "8":
        print('Main Menu \n'
              '1: Create a skyscraper\n'
              '2: Create a building\n'
              '3: Build a road, north to south\n'
              '4: Build a road, east to west\n'
              '5: Create a river, north to south\n'
              '6: Create a river, east to west\n'
              '7: Destroy a 50 cubed area around you\n'
              '8: Exit the menu')
        answer = input('What would you like to do? ')
```

Our final section of code is a conditional test, which will check the value stored in "answer", our variable that stores the player's choice. If the value stored matches one of the hard-coded values then it will run the respective function. For example if "answer" contained "4" then the correct choice would be to run **road('east')**. If the value stored in "answer" is not a number between 1 and 7, then it is highly likely the player chose "8", which will exit the menu by using the "else" condition. This will print "Goodbye" in the Python shell and print a message in the Minecraft chat window.

```
        if answer == '1':
                skyscraper()
        elif answer == '2':
                building()
        elif answer == '3':
                road('north')
        elif answer == '4':
                road('east')
        elif answer == '5':
                river('north')
        elif answer == '6':
                river('east')
        elif answer == '7':
                destroy()
        else:
                print('Goodbye')
                mc.postToChat('Exiting menu,
 connection terminated')
                break
```

And that is all of our code complete. Make sure you save your work before continuing.

In order to use our code, we first need to start a game of Minecraft – you can find it in the Games menu. With Minecraft open we now need to go back to the Python editor – but we can't, our mouse is stuck in Minecraft! Fear not, simply press the Tab key. This will release the mouse, and you can now click on your code. Click on Run > Run Module to start your code and then select an option from the menu to build your own city made of blocks!

To get back into Minecraft, click on the Minecraft window in the taskbar at the top of the screen, and then move your mouse. Minecraft should respond. If not, just click in the Minecraft game window and it will register your click.

Now go forth and build your own city – or destroy it! ■

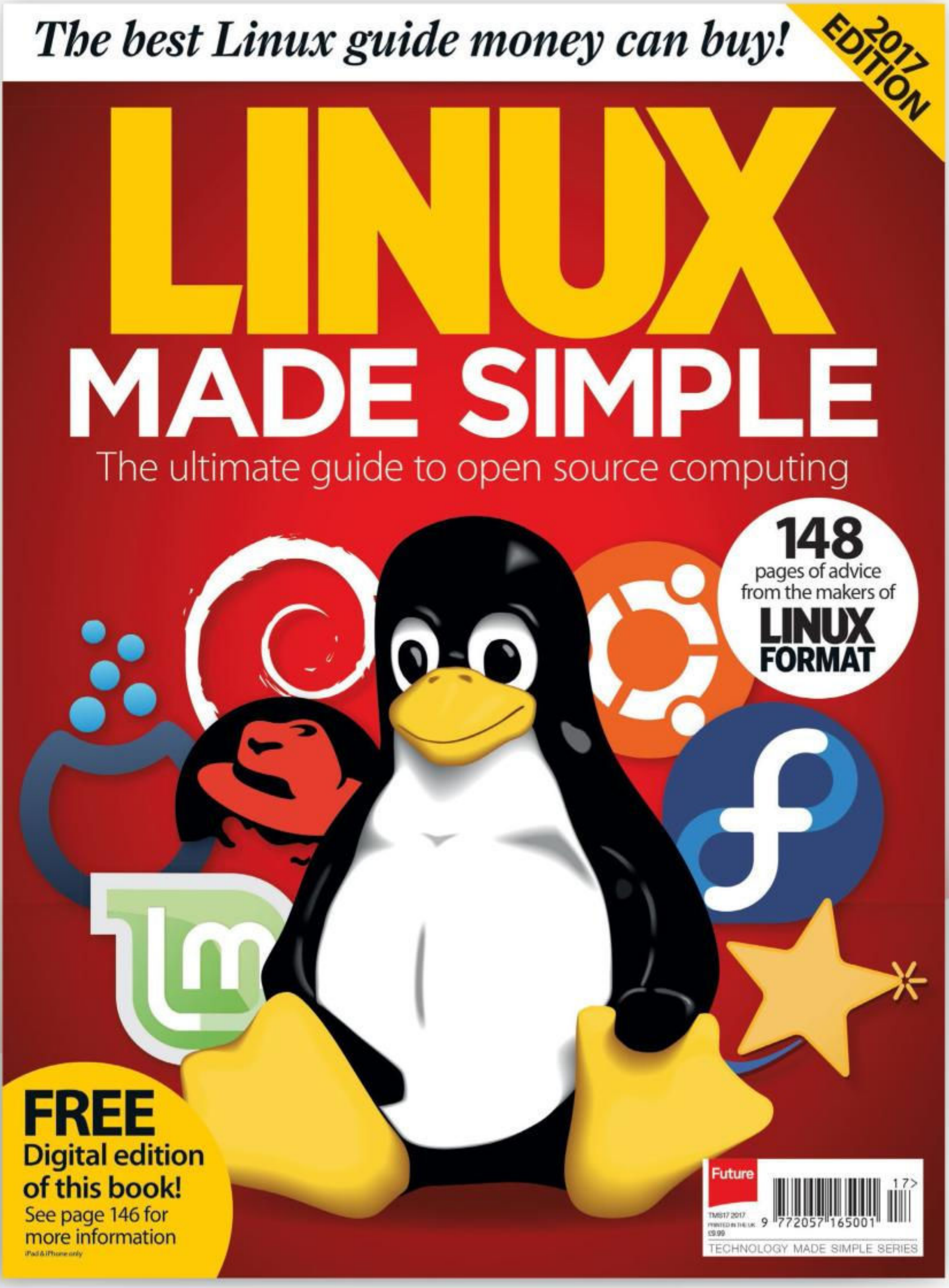
Pi bites

The "building" function is intentionally bare-bones. So to decorate your home, press the "e" key to open your inventory and select your furnishings.

MUSICAL MINECRAFT

We used Python in our project, but did you know that you can also hack Minecraft with Sonic Pi? Sonic Pi comes pre-installed on your Raspberry Pi and with it you can create music of any genre – and create Minecraft worlds. Sonic Pi was created by Dr Sam Aaron, a gifted computer scientist and musician. Sam uses Sonic Pi to "live code" compositions for events, rewriting the Ruby code live in front of an audience and reacting to the cues given by a live band. The Minecraft functionality was introduced into Sonic Pi as a result of a conversation between Sam and Martin O'Hanlon, an expert in Minecraft. As a result of this meeting we can use Minecraft as an input for a performance – for example changing music depending on the block that we are stood upon. You can read more at **http://sonic-pi.net**

GET STARTED WITH LINUX THE EASY WAY



The best Linux guide money can buy!

2017 EDITION

LINUX MADE SIMPLE

The ultimate guide to open source computing

148 pages of advice from the makers of LINUX FORMAT

FREE Digital edition of this book! See page 146 for more information

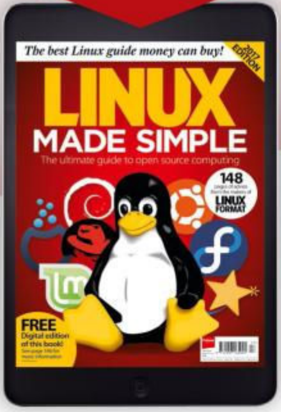
iPad & iPhone only

Future

TMS17 2017
PRINTED IN THE UK
£9.99

TECHNOLOGY MADE SIMPLE SERIES

OUT NOW! WITH FREE DIGITAL EDITION



Future

ORDER YOUR COPY TODAY!

Order online at **www.myfavouritemagazines.com**

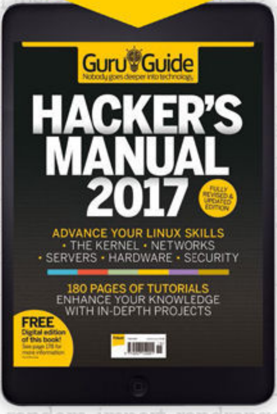
Also available in all good newsagents

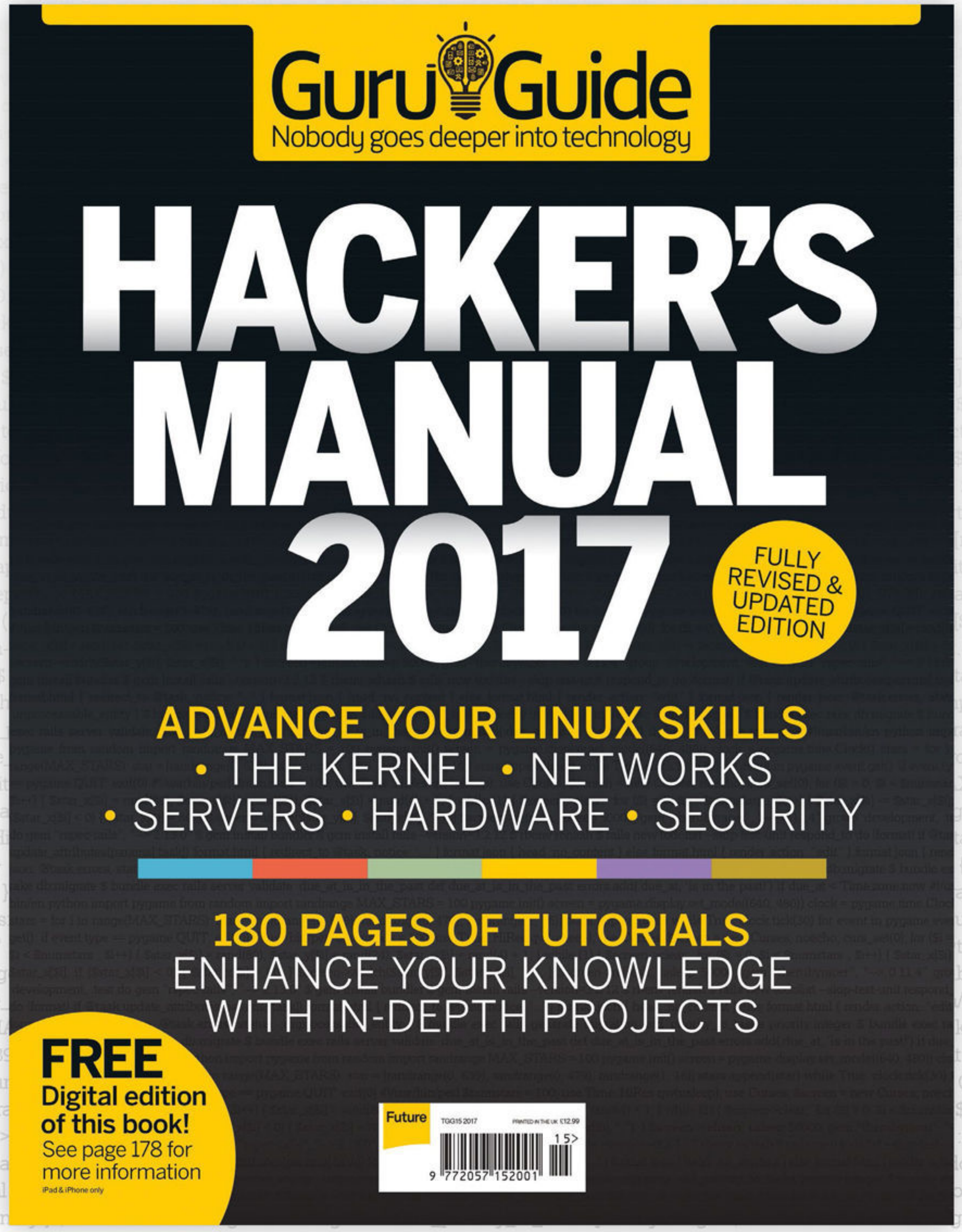
Available on the App Store

GET THE MAGAZINE ON Google play

TAKE YOUR LINUX KNOWLEDGE FURTHER

OUT NOW!
WITH FREE DIGITAL EDITION





Guru Guide
Nobody goes deeper into technology

HACKER'S MANUAL 2017

FULLY REVISED & UPDATED EDITION

ADVANCE YOUR LINUX SKILLS
• THE KERNEL • NETWORKS
• SERVERS • HARDWARE • SECURITY

180 PAGES OF TUTORIALS
ENHANCE YOUR KNOWLEDGE
WITH IN-DEPTH PROJECTS

FREE
Digital edition of this book!
See page 178 for more information
iPad & iPhone only

Future TGG15 2017 PRINTED IN THE UK £12.99
15>
9 772057 152001

DELIVERED DIRECT TO YOUR DOOR

Order online at **https://www.myfavouritemagazines.com**
or find us in your nearest supermarket, newsagent or bookstore!

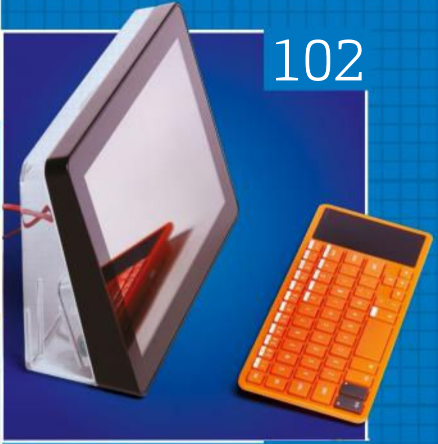
Pi REVIEWS

Tested and rated: the latest add-ons for your Pi

110

BEAUTIFULLY MADE AND ROBUST, THIS ADDS ETHERNET AND FULL-SIZE USB PORTS TO A PI ZERO

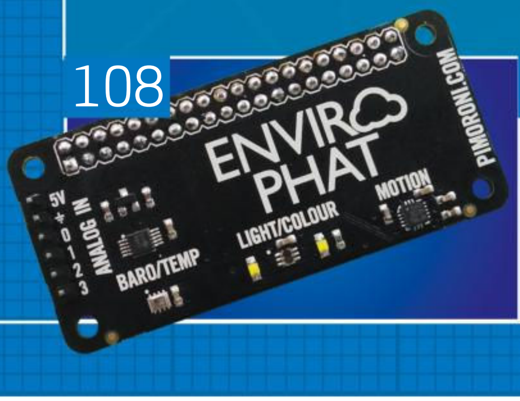
102



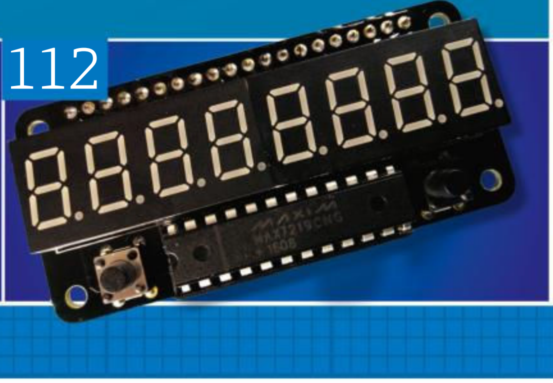
104

106

108



112



102 Kano Screen Kit

104 Pi-Top laptop kit

106 S.USV Pi Advanced

108 Enviro pHAT

110 Pi Zero RJ45 & USB Hub

112 ZeroSeg LED display kit

Kano Screen Kit

In his quest to create the ultimate Raspberry Pi portable hack station, **Les Pounder** finds a portable screen that will fit in his rucksack

In brief...

A portable 10.1-inch Raspberry Pi screen that offers HD resolution in a strong and kid-friendly chassis. The Kano ethos is to enable learners to experience building their own computer and this is supported by a series of workbooks written with children in mind.

New trends in Raspberry Pi accessories are like buses: you wait ages for a portable Raspberry Pi solution and then they all turn up at once. We've seen many different screens for the Raspberry Pi, including the official 7-inch screen, and the next one to turn up is this kit from Kano.

This company knows about packaging and marketing, and the Kano Screen Kit comes in a quality box with everything well protected and labelled. The box also includes an assembly guide written for young hackers, designed to be similar to Lego instructions, which enables users to start building as soon as the box is open.

The Kano screen measures 10.1 inches diagonally and is made with Gorilla Glass, which means it's well protected from knocks and scrapes. The screen comes as a kit which the user is encouraged to build. This kit has a strong plastic frame into which the screen is clipped, and the frame can be laid down on a table in a similar way to an old school desk or upright like a traditional monitor. On the back of the screen is a driver board that connects it to a Raspberry Pi via a conventional HDMI cable, and attached to the driver board is a breakout cable which attaches to a convenient screen controls board.

There's also a retention space to attach your Kano Raspberry Pi case to the screen, creating a rather nice single unit. The screen is powered via a bespoke micro USB cable that also powers your Raspberry Pi. Building the screen takes just a few minutes; the only tricky bit is attaching the driver board case to the screen.

> THERE'S NO SPEAKER, SO YOU'LL NEED TO ATTACH ONE, NOR DOES IT HAVE TOUCH INPUT, BUT THE PICTURE QUALITY IS EXCELLENT

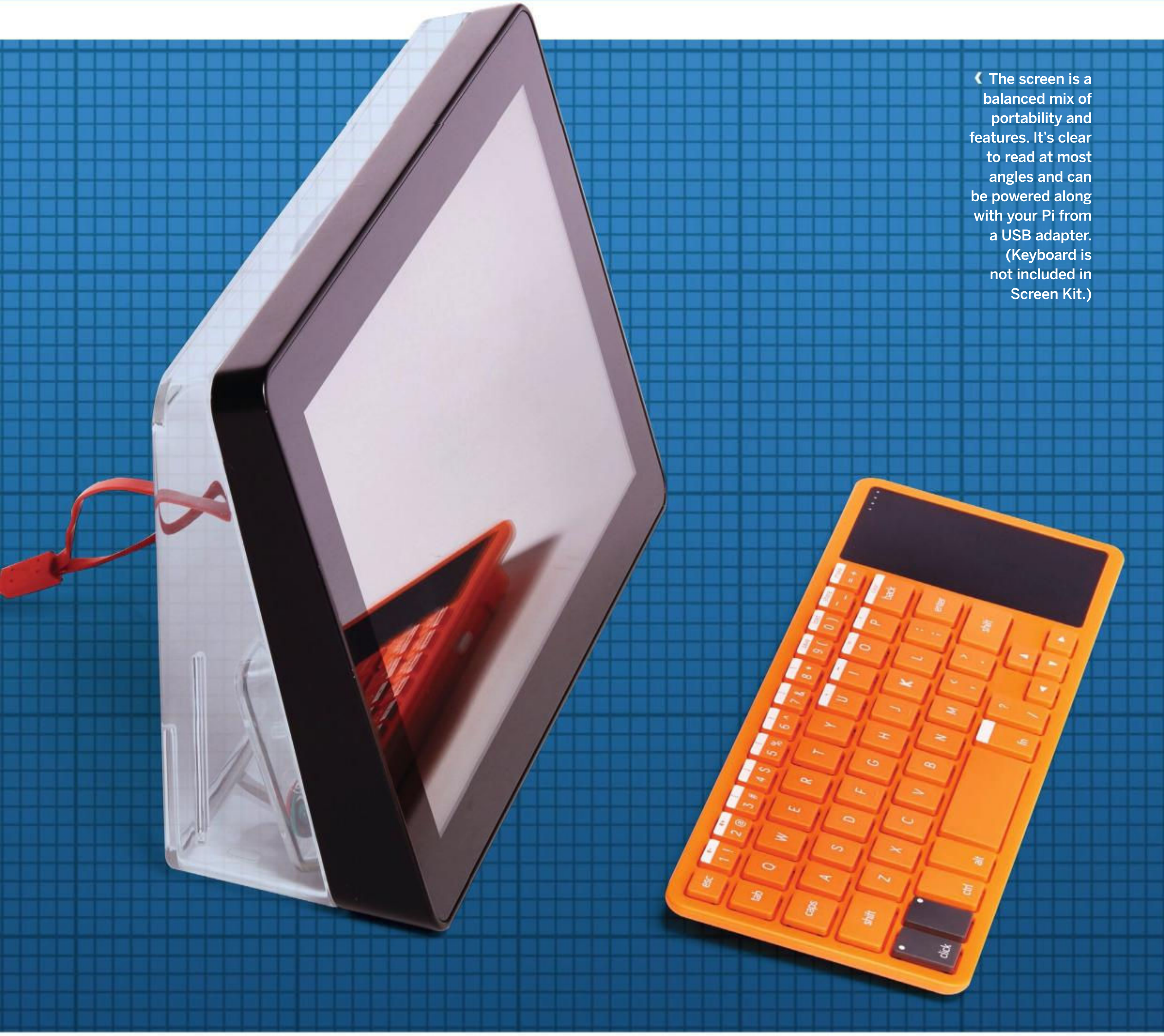
Tough contender

We tested the screen with the latest Kanux distro and the latest Raspbian image. For both screens the initial view appeared upside down, requiring us to lie the screen down so that it looked like an old school desk, and then rotate the screen 180 degrees in the OS. In Kanux this is achieved by going into the System Settings and selecting Display. In Raspbian you have to edit the **config.txt** file found in the boot partition and manually add `display_rotate=2` as the last line. For both distros the desktop was displayed at the correct size and no further configuration was necessary, but using the controls on the back of the screen we could quickly alter the setup of the screen if needed.

Features at a glance

Adjustable display
Provides two display positions and handy controls on the back to get your screen just right (and somewhere to stash a keyboard).

Works with all
The Kano screen uses an HDMI driver board, which means that it can also be used with other devices.



The screen is a balanced mix of portability and features. It's clear to read at most angles and can be powered along with your Pi from a USB adapter. (Keyboard is not included in Screen Kit.)

The screen does not provide any audio output, so you'll need to attach a speaker/headphones to the jack on your Raspberry Pi, nor does it have touch input. Picture quality is excellent, however, with the 10.1-inch screen running at 1,280x800, giving us 150 pixels per inch (PPI). The resulting image is bright and clear, providing the ideal platform for hacks.

The Kano Screen is a tough, well built and well specified product, and it's clear that the target market, young learners, is well catered for as this screen can take a fair amount of punishment, unlike other screens on the market.

The Kano Screen is currently priced at something of a sweet spot: it's not as cheap as the official screen and doesn't have a touchscreen capability, but it is built to last and right now it's a good alternative for users and schools when compared to other solutions, such as the Pi Top (see overleaf). The lack of audio output will be an issue for some buyers, especially those using the screen with the Pi Zero, which has no headphone socket. The Kano Screen is not a cheap screen solution, but it is one of the most robust at this price point. It successfully merges portability with usability, offering a lot of screen for a fair price. ■

Pi User Verdict

Kano Screen Kit

Developer: Kano
Web: kano.me
Price: £120

Rating 8/10

In brief

A Raspberry Pi laptop solution offering an impressive level of portability and features that are aligned with the UK GCSE curriculum. But these features come with a hefty price tag and a number of rough edges to a relatively new software base. That said, this is a powerful package for those who can work around the issues.

Pi-Top isn't quite your typical laptop, and under the lid it's all Raspberry Pi powered.

Features at a glance

Practical package

The Pi-Top laptop case provides easy access to the GPIO via a sliding panel above the keyboard.

Applications

Standard Raspbian apps are included, such as LibreOffice and the Chromium web browser.

Pi-Top laptop kit

Les Pounder gets hands-on with an education-focused Raspberry Pi laptop kit that promises to make the class green with envy…

Right now the Raspberry Pi sports a plethora of portable options. September 2015 saw the release of the official touchscreen, and other companies such as Kano offer their own portable (or near-portable) setups – see the previous page. Pi-Top started life as a crowd-funded project and combines both hardware and software. The hardware is a sturdy plastic laptop shell (supplied as a kit) and includes a 13.3-inch HD (1,366x768) LCD screen with eDP interface, an 8GB SD card, a battery with a claimed life of 10 hours, and a Wi-Fi dongle. You can buy it with or without a Pi 3. The kit is fairly simple to assemble but might require adult supervision for a few fiddly bits, such as attaching the LCD screen to the driver board.

FAIRLY SIMPLE TO ASSEMBLE BUT MAY REQUIRE ADULT HELP FOR A FEW FIDDLY BITS

The driver board handles connecting the Pi to the built-in battery, recharging the battery via an included external power supply, and sending HDMI video input to the LCD screen. Connecting the GPIO of your Pi to the driver board enables battery management, but will cover all 40 GPIO pins; it can be removed, enabling use of the GPIO. The driver board and the Pi (located at the right of the laptop) are covered by a slide cover, for quick access to the Pi. Access to the Pi's USB and Ethernet ports is tricky but possible. The keyboard and trackpad are fine for daily use but the keyboard can feel a little spongy and imprecise at times.

On the software side, you get the pi-topOS, built upon Raspbian Wheezy 7.8. Pi-topOS acts as a layer on top of the Raspbian OS. On first boot you are prompted to set up your Pi-Top, which includes creating an online account. This is used to save your learning progress to the Pi-Top cloud-based learning system, which is aligned with the UK Computing curriculum for 13 to 15 year olds.

Learning experience

Like Raspbian, the Pi-Top comes with a bundle of applications. These include Scratch, LibreWriter, Minecraft Pi, Sonic Pi and the Python editor IDLE. We tested IDLE 3, the Python 3 editor, with the RPi.GPIO library, the most popular library for hardware hackers and makers. Being based on Wheezy, IDLE3 was unable to access the GPIO using the default user – it required opening a terminal and running the command via **sudo** – but we successfully built and tested a simple LED project. This is a step backward for those used to the latest Raspbian Jessie image, but we'd expect this to be fixed in a future release. Of course, as it's based on Raspbian, you are free to install your favourite applications via the package manager.

During our tests there were a couple of issues. Logging in as an incorrect user prevented us from re-attempting a login with the correct details. A reboot solved this issue, but it did take time. Also, an update bug prevented pi-topOS from connecting to the update server despite constant reminders and using an Ethernet cable to connect to the router.

Pi-Top also has its own software, a game called CEED Universe, which teaches coding and making concepts via an interactive retro game – a great idea that keeps children engaged while learning key skills. It's reminiscent of Kano OS, which itself uses gamification to teach core concepts. CEED Universe is a lot of fun and provides a great level of interaction for children wanting to learn more.

The Pi-Top is a great idea and provides a portable Pi solution, but it's not as high-quality as the price might lead you to expect, and the cost of $300 (around £200) is high, with just a 30-day warranty, for its target education market. The software bugs will doubtless be tackled, but the cost is the main barrier. ■

Pi User Verdict

Pi-Top laptop kit

Developer: CEED Ltd
Web: www.pi-top.com
Price: $300 with a Pi 3, $265 without

Rating 6/10

S.USV Pi Advanced

Les Pounder has always loved Star Trek and often pretends to be Captain Kirk, but can Scotty provide more power? Maybe with this board...

In brief...

A flexible UPS (Uninterruptible Power Supply) that provides multiple power sources, such as batteries and solar power, for critical Raspberry Pi projects. The board attaches to the GPIO and comes with an extensive suite of software tools that enable configuration and management of even the most minute details of the board and the attached sources of power.

The Raspberry Pi has been used to power many different projects, including some where it is a mission-critical component. For these sorts of dedicated projects it is prudent to have a secondary means of powering the Pi, should the mains supply suffer a sudden interruption or shutdown.

An Uninterruptible Power Supply (UPS) provides a temporary solution to powering your Pi. The S.USV HAT is compatible with all versions of Raspberry Pi that come with the 40-pin GPIO and features a pass-through connector enabling the use of other Raspberry Pi add-on boards. The S.USV HAT, being a HAT compliant board, comes with an EPROM connected to pins 27 and 28 of the GPIO, which communicates with the Raspberry Pi and handles initial configuration. The rest of the configuration and installation is handled via the Terminal, and while being quite involved and requiring a number of changes to the default Raspberry Pi configuration, installation is straightforward and well covered in the supporting documentation. (Sadly the same cannot be said for all of the documentation – more on this shortly.)

The S.USV board connects and communicates with the Raspberry Pi using the I2C protocol, requiring only two GPIO pins for data, and an additional 5V and GND connection to power the board. We tested the Advanced version, which comes with a 300mAh (milli-Amp-hour) battery. A noticeable feature present only on the advanced model is the inclusion of an alternative power source which can be used with any power supply rated between 7V and 24V. The software that handles the functionality of the board is controlled by a background process, a daemon, which provides the options to start, stop and restart the daemon process. Installing the daemon also enables the use of two buttons, S1 (used to turn off or reboot your Pi) and S2 (which can be used to power up your Pi).

> THE CLIENT TOOL PROVIDES THE STATUS AND CONFIGURATION OF THE BOARD AND POWER SOURCE VIA A TERMINAL INTERFACE

Power user

The software is installed to ***/opt/susvd*** and comprises the aforementioned daemon and a client tool which can provide the status and configuration of the board and the attached power source via a handy terminal interface. The client tool has options to read and set the charging current for the battery, and we used it to ensure that the supplied 3.7V 300mAh battery was charged and ready for use. Also present are options to create shutdown timers and timed boots, using the clock embedded in the board, and the

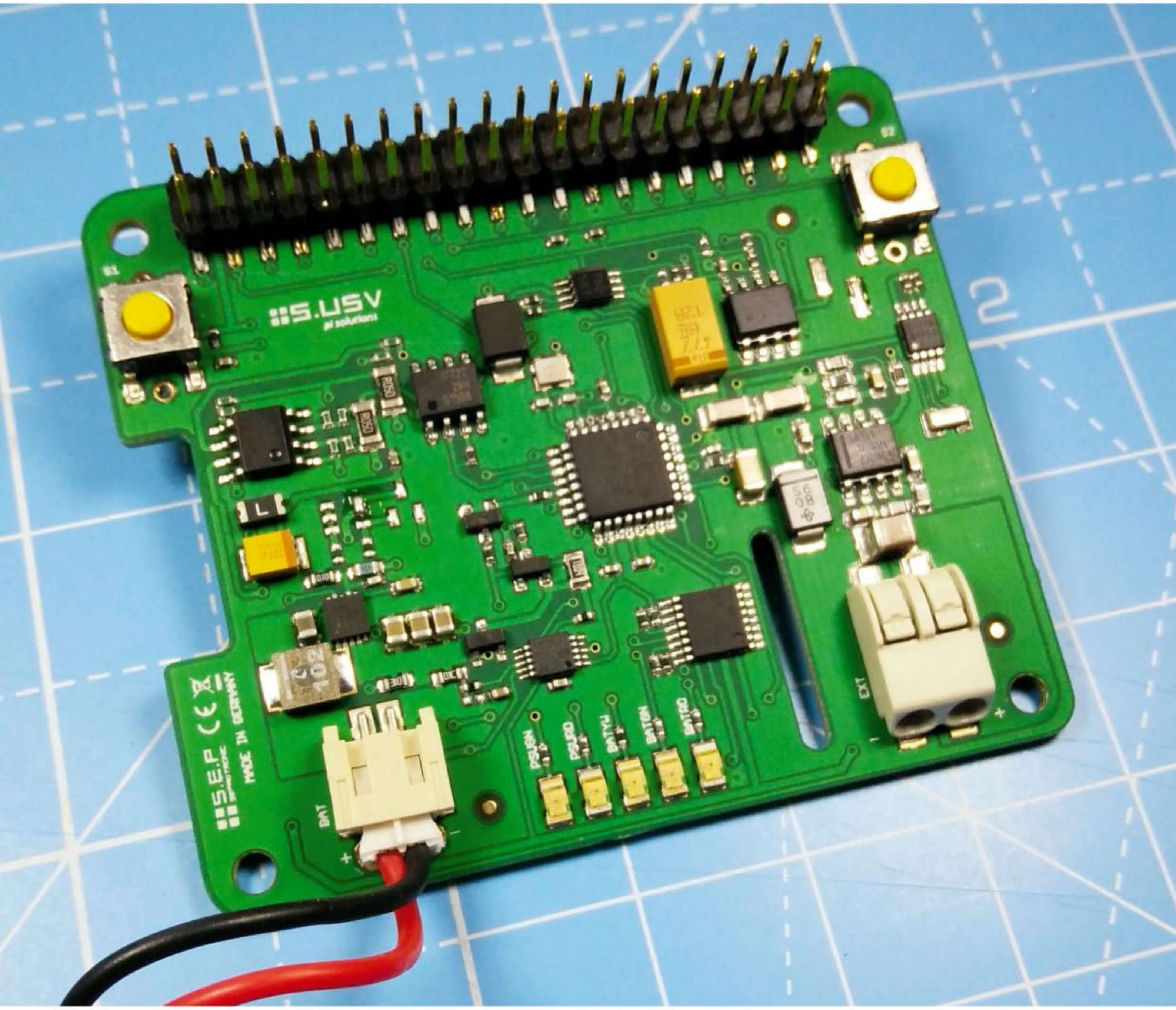
Features at a glance

Multiple power sources

The Advanced version (tested here) has inputs for two power sources – an LiPo battery plus other sources you can also connect as backups.

GPIO access

Some add-on boards prevent access to the GPIO but the handy pass-through enables access to it for direct connection or the use of other boards.



facility to upgrade the firmware to ensure that your board is running the latest version.

Sadly the documentation for this board has a few issues, chiefly found when trying to enable the real time clock (RTC). We found a few faults with the configuration necessary to enable the RTC, and this meant that we spent rather too much time to enable such a simple, yet key, piece of functionality. Hopefully this can be fixed in future revisions of the documentation.

This board promises so much, and it does deliver on the essentials of that promise – it just needs a little refinement to make it a better product. The documentation is generally great and it covers every aspect of installing and configuring the board; it is just let down by a few errors that cause "hiccups" for the user. On the plus side, this board provides full access to all of the GPIO pins – and since it uses I2C you can easily add more boards to your Raspberry Pi projects with little fear of conflicts.

This is a powerful UPS board that offers so much flexibility. It has a few documentation issues but is well worth considering. There are other UPS boards on the market, though, and when considering a purchase it would be prudent to look around for a board that provides exactly the functionality you require. ■

The S.USV board fits very neatly on top of all models of Raspberry Pi with 40 GPIO pins.

Pi User Verdict

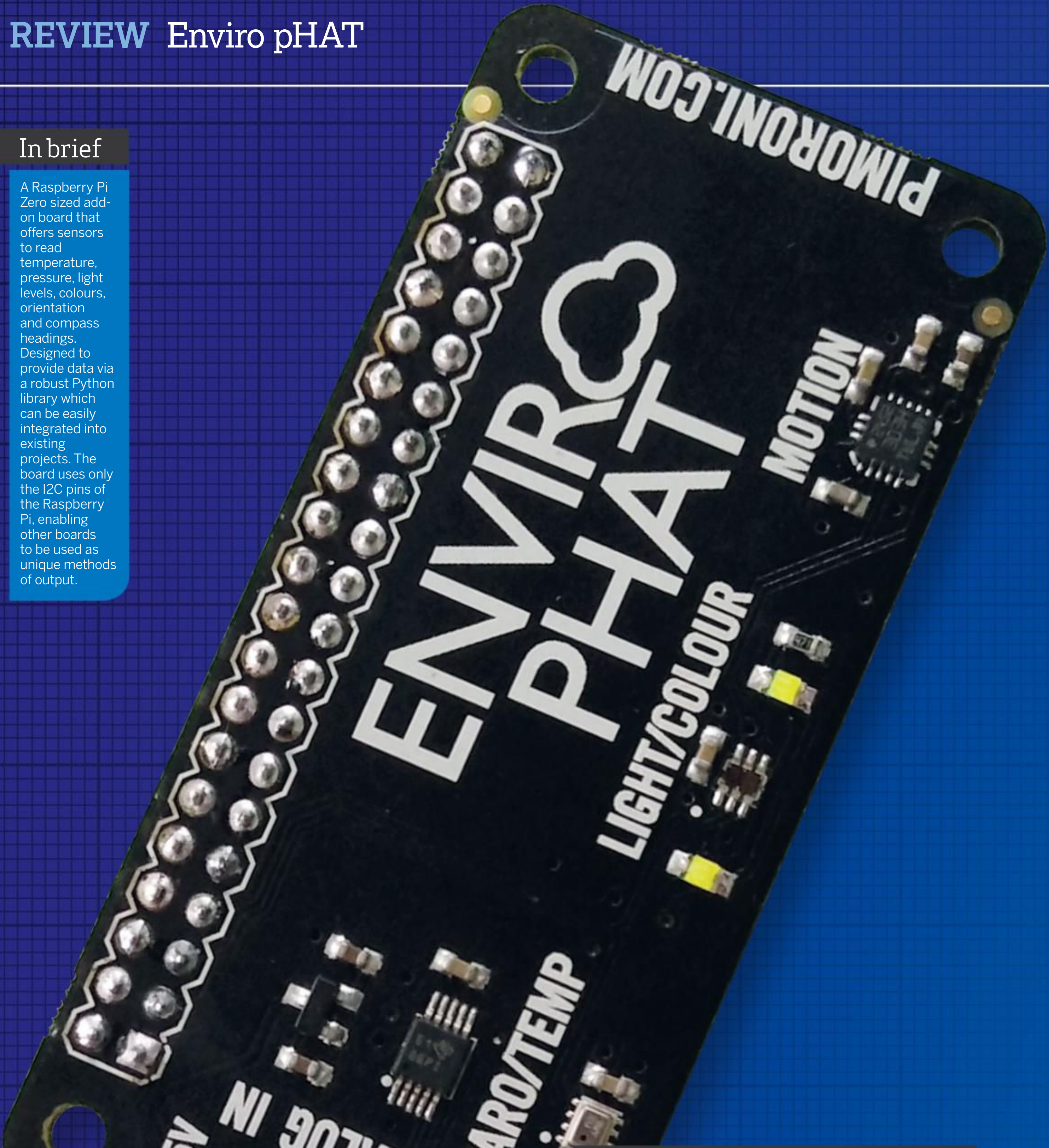
S.USV Pi Advanced

Developer: SEProtronic GmbH
Web: www.s-usv.de/index_en.php
Price: Basic €29.99 (about £26), Advanced (tested) €54.99 (about £48)

Rating 7/10

In brief

A Raspberry Pi Zero sized add-on board that offers sensors to read temperature, pressure, light levels, colours, orientation and compass headings. Designed to provide data via a robust Python library which can be easily integrated into existing projects. The board uses only the I2C pins of the Raspberry Pi, enabling other boards to be used as unique methods of output.


PIMORONI.COM
ENVIRO PHAT
MOTION
LIGHT/COLOUR
BARO/TEMP
ANALOG IN
5V
+
0
1
2
3

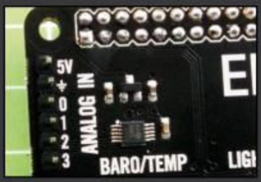
The Enviro pHAT is a compact sensor board designed to fit atop the Raspberry Pi Zero and sit with a flush profile, enabling small, neat projects.

Features at a glance

ENVIRO PHAT
LIGHT/COLOUR
MOTION

Sensors

The Enviro pHAT has sensors for temperature/pressure, light/colour and motion/orientation. These are simple to integrate into any Python project.


5V
+
0
1
2
3
ANALOG IN
BARO/TEMP

Analog input

The Raspberry Pi GPIO does not have an analog input by default. The Enviro pHAT comes with its own ADC for extending the number of sensors.

Enviro pHAT

Les Pounder loves data and is always looking for new ways to integrate real-time data into his projects. Perhaps this board can fulfil his needs?

The Pirates of Sheffield bring us another board using their own pHAT standard for smaller add-on boards primarily designed for the compact Raspberry Pi Zero but compatible with all Raspberry Pi boards with 40 GPIO pins. It's also worth noting that the Enviro pHAT uses only three I2C pins (I2C being a simplified communication protocol) and two other pins for power, which means it is possible to attach the board to older Pis using jumper cables or a breakout board.

The Enviro pHAT is a platform for data gathering, similar to the Sense HAT used for the Astro Pi project. The Enviro pHAT comes with a plethora of sensors. The BMP280 temperature and pressure sensor can work between temperatures of −40C and +85C, and pressures of 300 to 1100 hPa (hecto Pascals). The TCS3472 light and RGB colour sensor can provide a reading for the light level, enabling it to be used as a trigger for light dependent projects, and can also identify colours, returning the value detected as a comma separated list of values, known as a tuple. The sensor's accuracy can be increased by illuminating objects using two white LEDs on each side of the light sensor. Next is an LSM303D accelerometer and magnetometer. The accelerometer detects orientation and motion of the board. A basic compass heading can be taken using the magnetometer. So with this single sensor we can create an input based on the orientation or heading of the board, handy for gesture controlled projects.

> THE BOARD HAS A PLETHORA OF SENSORS AND A FOUR-CHANNEL ADC FOR CONNECTING MORE

As well as sensors the board comes with an ADS1015 four-channel analog-to-digital converter (ADC) that can be used with external analog sensors. Attaching a sensor to the ADC is rather simple if it uses 3.3V logic, requiring that we simply attach the sensor to the header pins present on the board. If your sensor uses 5V logic, and this can be identified in the data sheet for the sensor, then you will need to use a voltage divider, commonly three resistors of equal value.

Assembling the Enviro pHAT requires some basic soldering skills. Controlling the Enviro pHAT is then handled via a robust and easy to use Python 2 / 3 library that can be used with all of the sensors, or individual sensors can be used by importing each class as needed. Installation of the Python libraries is automated thanks to an install script available from the Pimoroni website. Typically using install scripts from random websites is not the done thing, but in this case we can trust the source. For those who wish to install manually there are full instructions and a step-by-step guide.

The Enviro pHAT comes into its own in a data logging project. Each of the sensors can be polled and data recorded into an external file, such as a CSV file, which can then be imported into a spreadsheet application, or the data can be used with an online resource such as plot.ly.

As mentioned earlier, the board uses only the I2C pins of the GPIO, meaning that other boards can also be connected. For example you can use the Enviro pHAT to gather data and then display that data via a chain of WS2811 LEDs, commonly known as Neopixels, where different colours denote different weather conditions or react to severe accelerometer input.

The Enviro pHAT is a great board that can be easily integrated into a home automation project or an experiment. The build quality is exceptional and the supporting Python libraries are easy to use while providing an extensive range of data capture options for those who wish to take on advanced projects. A cost-effective and easy-to-use add-on board for gathering real-time data about your environment. ■

Pi User Verdict

Enviro pHAT

Developer: Pimoroni
Web: http://bit.ly/enviro-pHAT
Price: From £16

Rating 8/10

Pi Zero Ethernet RJ45 and USB Hub

Les Pounder tests a community-produced board that provides 100Mbps Ethernet and full-size USB ports for the Pi Zero. But is it worth the price?

In brief...

A community made project that enables full-size USB and Ethernet ports to be added to the Pi Zero. A solid PCB and no configuration enable this board to be used for permanent projects, or used temporarily for configuring a project. Pogo pins provide a temporary yet stable solderless connection between board and Pi Zero.

When the Raspberry Pi Zero first came out, many applauded the reduced cost but some were unhappy with the micro USB port and lack of Ethernet. Many different community boards were created to address this issue, including this one. The Raspberry Pi Zero Ethernet RJ45 and USB Hub Shield is a project created by an individual maker in Scotland and available for sale via the open hardware marketplace Tindie.com.

It's created using a four-layer printed circuit board and has been made with great attention to detail. On the board, there are two USB 2.0 ports mounted on their side with sufficient clearance for cumbersome USB connectors to be used. Next to the USB ports is a single Ethernet port rated for 10/100. All of the USB and Ethernet ports are routed via a LAN9512 USB control chip – the same chip used on all Pi models prior to the release of the Raspberry Pi B+ in 2014.

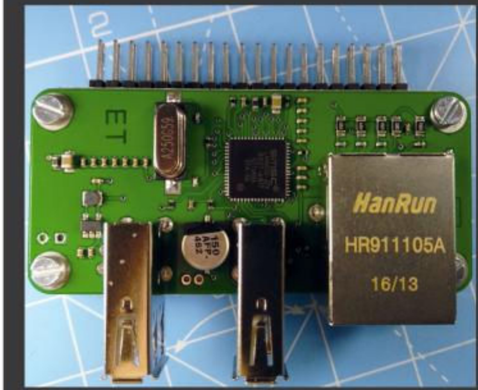
> NO SOFTWARE INSTALLATION OR CONFIGURATION IS REQUIRED – DRIVERS FOR THE CONTROL CHIP ARE ALREADY IN RASPBIAN

The board attaches to the Pi Zero using pogo pins that lightly connect to the USB and power test pads present on the underside of the board (PP1, PP6, PP22 and PP23). The board is compatible only with v1.3 of the Pi Zero, released in early 2016. This is because of the placement of PP1 and PP6 being different on the previous version, 1.2. With any luck, this shouldn't matter to many users, considering that v1.3 has now been with us for much longer than the short-lived 1.2. With the board attached to the underside of your Pi Zero you still have access to all 40 of the GPIO pins. The board is secured using the included fastening kit, which we found a little fiddly but ultimately not that difficult to install.

Zero effort

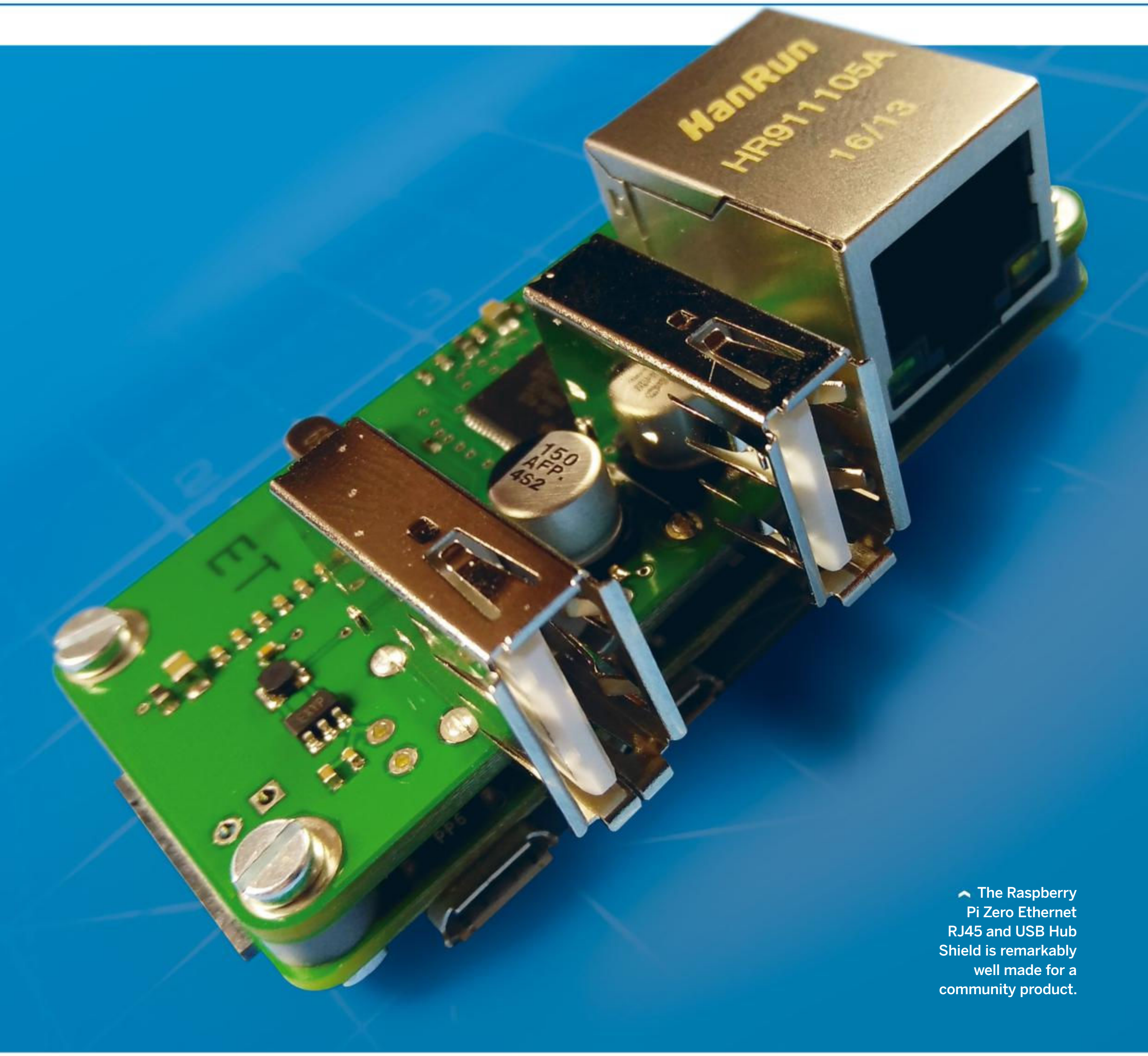
Once the board is attached, all you need to do is connect your accessories and power up using the Pi Zero micro USB power port. Here is where we hit another issue. Once the board is attached, the Pi Zero's own micro USB port, normally used for keyboards and mice etc, is now non-functional and all connections have to be made using the add-on board. This isn't a deal breaker by any means but it would have been nice to have three USB ports, even if one was micro USB.

Features at a glance



USB and Ethernet
The LAN9512 USB/Ethernet controller chip is used on all models of Pi up to the B+. As a result, it works out-of-the-box with no additional configuration required.

Pogo pins
Use light pressure to create a temporary contact between the board and the gold test pads on the underside of the Pi Zero.



The Raspberry Pi Zero Ethernet RJ45 and USB Hub Shield is remarkably well made for a community product.

On the positive side, there's no software installation or configuration needed, largely thanks to the LAN9512 USB control chip, for which drivers are already present in Raspbian. For advanced Pi users there are breakout points for the 5V and GND pins present on the GPIO. This means that power can be provided using an alternative connection to the micro USB.

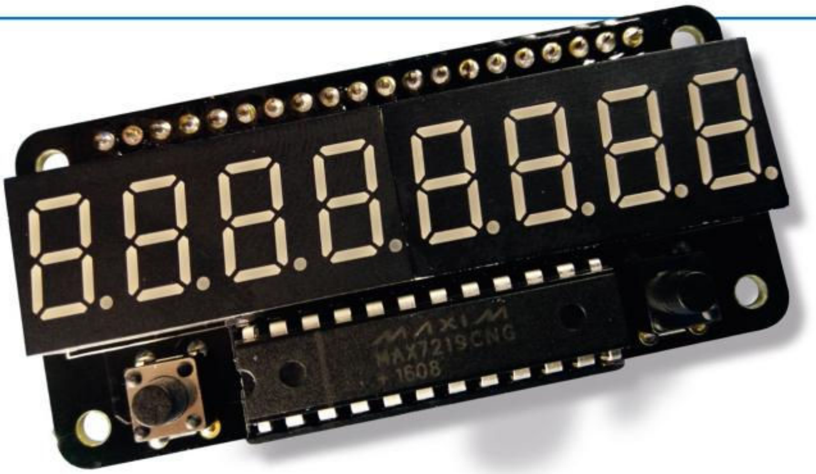
Beautifully made and robust, this board isn't cheap, but it does deliver an effective and elegant way to add Ethernet and full-size USB ports to your Pi Zero. The fact that it's compatible only with the current v1.3 of the Pi Zero should not be a real drawback, but something to be aware of for any potential purchaser. The price might however be an issue. Because this is made by an individual, the cost of each board is higher than you'd expect of boards mass-produced in a factory. Adding on the cost of shipping, plus the initial cost of a Pi Zero, takes us over £20 and into Pi 2 and 3 territory… But if you really need to use a Pi Zero in your project and you need USB and Ethernet, then this is the board for you. ■

Pi User Verdict

Raspberry Pi Zero Ethernet RJ45 and USB Hub Shield

Developer: Marcel
Web: http://bit.ly/PiZeroEthandUSBHub
Price: $22 (about £18)

Rating 9/10



The ZeroSeg kit is fun to build. Sure, there's a few tricky bits of soldering involved, but that's part of the challenge.

ZeroSeg

Les Pounder looks at a kit that can be used to add a classic '80s display to your Pi

In brief...

A dual seven-segment LED display controlled using a MAX7219 SPI chip and a straightforward Python library, which presents a humble yet powerful platform for makers. As a kit that requires some soldering, the board offers a cost-effective entry for those interested in using the display and its two tactile switches to control a headless Raspberry Pi project.

Growing up in the '80s, we remember the humble seven-segment display that flashed "12:00" on our family's first VCR. This simple method of output provided enough information to use the VCR, from just seven LEDs per character. In the 21st century we have OLED displays and basic holographic displays, but there's still a place for the seven-segment display.

The ZeroSeg is an add-on board for all 40-pin Raspberry Pi models but is designed primarily for the Pi Zero and fits neatly atop it. The board was originally designed by a community member, AverageManVsPi, who has started a small range of boards aimed at the beginner. The ZeroSeg comes as a self-assembly kit and requires soldering. While this is mostly trivial there are a few moments, such as soldering the displays next to the plastic housing of the female GPIO header, where it can become a little tricky. Steady hands and good eyesight are required.

The board has two four-digit seven-segment displays (so eight characters in all) and two tactile switches. The displays are controlled using a MAX7219CNG chip designed for driving 8-bit LED displays via an SPI interface – something the Pi is capable of using for communication. This common chip is also used to drive 8x8 grids of LEDs for digital signage and it's nice to see that it's housed in a holder, allowing replacement should the need arise. To use the board you need to install a number of drivers and libraries, and all this is well documented in a concise PDF download, which also includes assembly instructions.

WITHIN A FEW MINUTES WE HAD TEXT SCROLLING ACROSS THE DISPLAYS, THOUGH NOT @ OR W

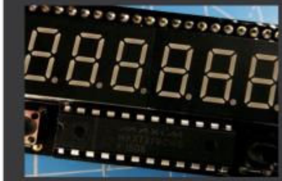
Neat display

With the set-up complete, we can start using ZeroSeg. The Python library for this board is a fork of Richard Hull's MAX7219 library, tweaked for ZeroSeg use. The library is Python 2 only, though; it would be better to see a Python 3 library as standard.

The tactile switches can be used with RPi.GPIO and GPIO Zero Python libraries, and we hear whispers of seven-segment display compatibility being added to GPIO Zero, enabling even easier hacks. The example scripts provide enough of a primer for you to get to grips with the board. They address common use cases such as scrolling text, simple date, time and temperature data.

Within a few minutes we were scrolling text across the displays and we hit a little snag: "@" or "w" cannot be displayed. This is no fault of AverageManVsPi, but a design limitation of seven-segment displays. This apart, for projects where you need simple output – say a camera timer, system monitor or IoT coffee pot monitor – this board is ideal, and at only £10 it's a great opportunity to dust off your soldering iron and build your own kit. It might not do anything unique – you could just build your own seven-segment display on a breadboard – but this is a fun, inexpensive board providing a functional (although limited) method of output that can be used in many different projects. If you've never used a seven-segment display, try it and learn how our old VCR talked to us. ■

Features at a glance



MAX7219
The common chip is used to drive the dual seven-segment display and it connects to the Pi using an SPI interface.

Design
Careful thought has led to small components being on the back and their solder joints covered by the larger, front facing parts.

Pi User Verdict

ZeroSeg

Developer: AverageManVsPi & Pi Hut
Web: http://bit.ly/ZeroSeg
Price: £10

Rating 8/10

Learn and make with Raspberry Pi

Get creative with your Raspberry Pi as we reveal even more home projects for you to try, and the latest apps.

Projects

Run a security camera, upgrade your Pi and more!

Coding

We continue our series of tutorials on Scratch and Python and show you even more Minecraft hacks.

Reviews

Discover the latest gadgets to go with your Pi.

*Contents of future issues subject to change

Future Publishing, Quay House,
The Ambury, Bath, BA1 1UA, UK
Tel +44 (0)1225 442244
Email piuser@futurenet.com

EDITORIAL
Editor-in-Chief Graham Barlow
graham.barlow@futurenet.com
Production editor Alex Summersby
Art editor Nick Aspell
Editorial contributors Jonni Bidwell, Les Pounder, Mayank Sharma, Nate Drake, Nick Peers, Gary Marshall

ADVERTISING
Commercial sales director Clare Dove
clare.dove@futurenet.com
Senior advertising manager Lara Jaggon
lara.jaggon@futurenet.com
Advertising manager Michael Pyatt
michael.pyatt@futurenet.com
Director of agency sales Matt Downs
matt.downs@futurenet.com
Ad director – Technology John Burke
john.burke@futurenet.com
Head of strategic partnerships Clare Jonik
clare.jonik@futurenet.com

MARKETING
Marketing director Sascha Kimmel
sascha.kimmel@futurenet.com

PRODUCTION AND DISTRIBUTION
Production controller Nola Cokely
Head of production UK & US Mark Constance
Distributed by Seymour Distribution Ltd, 2 East Poultry Avenue, London EC1A 9PT **Tel** 020 7429 4000
Overseas distribution by Seymour International

LICENSING
Senior Licensing & Syndication Manager Matt Ellis
matt.ellis@futurenet.com **Tel** + 44 (0)1225 442244

CIRCULATION
Trade marketing manager Juliette Winyard
Tel 07551 150 984

THE MANAGEMENT
Editorial director Paul Newman
Group art director Graham Dalzell

Raspberry Pi is a trademark of the Raspberry Pi Foundation. See **www.raspberrypi.org**

LINUX is a trademark of Linus Torvalds. GNU/Linux is abbreviated to Linux throughout for brevity. All other trademarks are the property of their respective owners. Where applicable code printed in this magazine is licensed under the GNU GPL v2 or later. See **www.gnu.org/copyleft/gpl.html**.

Entire contents copyright © 2017 Future Publishing Ltd or published under licence. No part of this publication may be reproduced without written permission from the publisher. We assume all letters sent – by email, fax or post – are for publication unless otherwise stated, and reserve the right to edit contributions. All contributions to *Pi User* are submitted and accepted on the basis of non-exclusive worldwide licence to publish or license others to do so unless otherwise agreed in advance in writing. *Pi User* recognises all copyrights in this issue. Where possible, we have acknowledged the copyright holder. Contact us if we haven't credited your copyright and we will always correct any oversight. We cannot be held responsible for mistakes or misprints.

Disclaimer All tips in this magazine are used at your own risk. We accept no liability for any loss of data or damage to your computer, peripherals or software through the use of any tips or advice.

Printed in the UK by William Gibbons on behalf of Future.

Distributed in the UK by Marketforce, 2nd Floor, 5 Churchill Place, Canary Wharf, London, E14 5HU

Future

Future is an award-winning international media group and leading digital business. We reach more than 57 million international consumers a month and create world-class content and advertising solutions for passionate consumers online, on tablet & smartphone and in print.

Future plc is a public company quoted on the London Stock Exchange (symbol: FUTR).
www.futureplc.com

Chief executive officer Zillah Byng-Thorne
Non-executive chairman Peter Allen
Chief financial officer Penny Ladkin-Brand
Managing director, Magazines Aaron Asadi
Tel +44 (0)1225 442 244

We are committed to only using magazine paper which is derived from well-managed, certified forestry and chlorine-free manufacture. Future Publishing and its paper suppliers have been independently certified in accordance with the rules of the FSC (Forest Stewardship Council).

recycle
When you have finished with this magazine please recycle it.

FSC
www.fsc.org
MIX
Paper from responsible sources
FSC® C007184

The Picademy

The Raspberry Pi Foundation rolls out more free training for educators in 2017

Some people, possibly teachers, in Manchester, behaving perfectly normally.

Image credit : Raspberrypi.org

The Raspberry Pi was created to help children learn programming skills from a young age, so it's no surprise that the Raspberry Pi Foundation has always had close links to schools and education. In fact, the Code Club concept (**www.codeclub.org.uk**) – where an enthusiast teams up with a school or library to run an after-school club for kids aged 9-11 using specially created resources – was a big hit last year. There are now 75,000 children in the UK regularly attending 5,000 code clubs.

Another of the initiatives the Pi Foundation have set up recently to help educators get to grips with the Pi is the Picademy, a free CPD (Continuing Professional Development) programme that aims to give teachers the skills and knowledge they need to use a Raspberry Pi in their place of work, no matter what level of computing experience they have.

PICADEMY GIVES TEACHERS THE SKILLS AND KNOWLEDGE TO USE A PI, WHATEVER THEIR COMPUTING EXPERIENCE

2016 was a great year for the Picademy, with 540 people graduating the courses, but the Raspberry Pi Foundation want 2017 to be even better. They have just released details of the courses available for 2017. They are at The TramShed in Cardiff on 21/22 February, the MadLab in Manchester on 14/15 March and 2/3 October, the Learning Hub @ Birmingham Airport 10/11 April and 4/5 December, and The Raspberry Pi Foundation in Cambridge 15/16 May. There will also be dates in May and November in London at a location to be confirmed.

A Picademy course lasts two days, and is a mixture of digital making workshops, project-based learning and hacking. The Pi Foundation are keen to stress that you don't need to have had any previous experience to attend the Picademy, just a love of computing and an enthusiasm for teaching it to kids. So, if you work in education and think you'd benefit from the course, then head to **https://www.raspberrypi.org/picademy/picademyuk/apply** and fill in the application form.

The only bad news is that the courses tend to be over-subscribed, so make sure you spend a decent amount of time on your application, giving them the chance to see who you really are and why the course would benefit you. ■

Future

lifehacker | UK

Helping you live better & work smarter

lifehacker | UK

SEARCH FOLLOW

Future

LIFE WORK BUSINESS TECH

3 easy steps for faster PC

Start Now

1. Click Start Now
2. Run the quick scan
3. Fix the errors.

- Regclean Pro

PLANHACKER

Planhacker: All the Best Samsung Galaxy Note 4 UK Deals and Tariffs

By Dan Grabham on 29 Sep 2014 at 1:30PM

Samsung's latest flagship productivity phone includes some super cool new features - here's all the best Note 4 UK plans and tariffs so far.

FEATURED

ANNOUNCEMENTS

IOS

DOWNLOADS

Killing this app now will save 40m 40s ± 2m 34s of battery (99% confidence).

3 easy steps for faster PC

LIFEHACKER UK IS THE EXPERT GUIDE FOR ANYONE LOOKING TO GET THINGS DONE

- Thousands of tips to improve your home & workplace
- Get more from your smartphone, tablet & computer
- Be more efficient and increase your productivity

www.lifehacker.co.uk

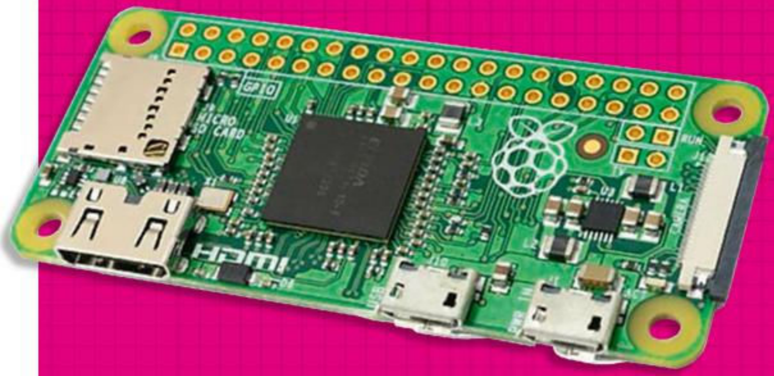
twitter.com/lifehackeruk facebook.com/lifehackeruk

PiUser

YOUR NEW SOURCE FOR RASPBERRY PI
JOIN THE PI
REVOLUTION

Discover the world of Raspberry Pi

The latest Pi projects, news and reviews



Perfect for owners of all models of the amazing Raspberry Pi and the fantastic Pi Zero

116
PAGES OF Pi GOODNESS INSIDE!

LEARN HOW TO...

P22

Build a
SUPER Pi!

Improve and enhance your Raspberry Pi

P61

Build a Pi-powered Wi-Fi walkie-talkie!

P86

Code with Scratch, Python and Minecraft